AD-A285 844

AGARD-AG-325

AGARD-AG-325

AGARD

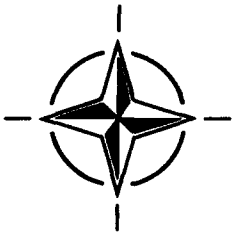
ADVISORY GROUP FOR AEROSPACE RESEARCH & DEVELOPMENT
7 RUE ANCELLE, 92200 NEUILLY-SUR-SEINE, FRANCE

AGARDograph 325

Computational Aerodynamics Based on the Euler Equations

(L'aérodynamique numérique à partir des équations d'Euler)

This AGARDograph has been produced at the request of the Fluid Dynamics Panel (FDP) of AGARD.



NORTH ATLANTIC TREATY ORGANIZATION

Published September 1994

Distribution and Availability on Back Cover

AGARD

ADVISORY GROUP FOR AEROSPACE RESEARCH & DEVELOPMENT
7 RUE ANCELLE, 92200 NEUILLY-SUR-SEINE, FRANCE

AGARDograph 325

Computational Aerodynamics Based on the Euler Equations

(L'aérodynamique numérique à partir des équations d'Euler)

Edited by

J.W. SLOOFF
National Aerospace Laboratory NLR
Anthony Fokkerweg 2
1059 CM Amsterdam
Netherlands

and

Dr. W. SCHMIDT
Air Vehicle Engineering
Military Aircraft Division
Deutsche Aerospace
81663 Munchen
Germany

DTIC QUALITY INSPECTED 2

This AGARDograph has been produced at the request of the Fluid Dynamics Panel (FDP) of AGARD.

94-33412



North Atlantic Treaty Organization
Organisation du traité de l'Atlantique Nord

9410 27 131

The Mission of AGARD

According to its Charter, the mission of AGARD is to bring together the leading personalities of the NATO nations in the fields of science and technology relating to aerospace for the following purposes:

- Recommending effective ways for the member nations to use their research and development capabilities for the common benefit of the NATO community;
- Providing scientific and technical advice and assistance to the Military Committee in the field of aerospace research and development (with particular regard to its military application);
- Continuously stimulating advances in the aerospace sciences relevant to strengthening the common defence posture;
- Improving the co-operation among member nations in aerospace research and development;
- Exchange of scientific and technical information;
- Providing assistance to member nations for the purpose of increasing their scientific and technical potential;
- Rendering scientific and technical assistance, as requested, to other NATO bodies and to member nations in connection with research and development problems in the aerospace field.

The highest authority within AGARD is the National Delegates Board consisting of officially appointed senior representatives from each member nation. The mission of AGARD is carried out through the Panels which are composed of experts appointed by the National Delegates, the Consultant and Exchange Programme and the Aerospace Applications Studies Programme. The results of AGARD work are reported to the member nations and the NATO Authorities through the AGARD series of publications of which this is one.

Participation in AGARD activities is by invitation only and is normally limited to citizens of the NATO nations.

The content of this publication has been reproduced
directly from material supplied by AGARD or the authors.

Published September 1994

Copyright © AGARD 1994
All Rights Reserved

ISBN 92-836-1005-9



Printed by Canada Communication Group
45 Sacré-Cœur Blvd., Hull (Québec), Canada K1A 0S7

Recent Publications of the Fluid Dynamics Panel

AGARDOGRAPHS (AG)

Scale Effects on Aircraft and Weapon Aerodynamics
AGARD AG-323, July 1994

Design and Testing of High-Performance Parachutes
AGARD AG-319, November 1991

Experimental Techniques in the Field of Low Density Aerodynamics
AGARD AG-318 (E), April 1991

Techniques Expérimentales Liées à l'Aérodynamique à Basse Densité
AGARD AG-318 (FR), April 1990

A Survey of Measurements and Measuring Techniques in Rapidly L[illegible]essible Turbulent Boundary Layers
AGARD AG-315, May 1989

REPORTS (R)

Missile Aerodynamics
AGARD R-804, Special Course Notes, June 1994

Progress in Transition Modelling
AGARD R-793, Special Course Notes, April 1994

Shock-Wave/Boundary-Layer Interactions in Supersonic and Hypersonic Flows
AGARD R-792, Special Course Notes, August 1993

Unstructured Grid Methods for Advection Dominated Flows
AGARD R-787, Special Course Notes, May 1992

Skin Friction Drag Reduction
AGARD R-786, Special Course Notes, March 1992

ADVISORY REPORTS (AR)

Quality Assessment for Wind Tunnel Testing
AGARD AR-304, Report of WG15, July 1994

Air Intakes for High Speed Vehicles
AGARD AR-270, Report of WG13, September 1991

Appraisal of the Suitability of Turbulence Models in Flow Calculations
AGARD AR-291, Technical Status Review, July 1991

Rotary-Balance Testing for Aircraft Dynamics
AGARD AR-265, Report of WG11, December 1990

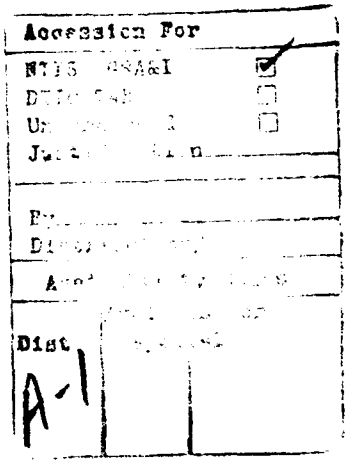
Calculation of 3D Separated Turbulent Flows in Boundary Layer Limit
AGARD AR-255, Report of WG10, May 1990

CONFERENCE PROCEEDINGS (CP)

Wall Interference, Support Interference and Flow Field Measurements
AGARD CP-535, July 1994

Computational and Experimental Assessment of Jets in Cross Flow
AGARD CP-534, November 1993

High-Lift System Aerodynamics
AGARD CP-515, September 1993



Theoretical and Experimental Methods in Hypersonic Flows
AGARD CP-514, April 1993

Aerodynamic Engine/Airframe Integration for High Performance Aircraft and Missiles
AGARD CP-498, September 1992

Effects of Adverse Weather on Aerodynamics
AGARD CP-496, December 1991

Manoeuvring Aerodynamics
AGARD CP-497, November 1991

Vortex Flow Aerodynamics
AGARD CP-494, July 1991

Missile Aerodynamics
AGARD CP-493, October 1990

Aerodynamics of Combat Aircraft Controls and of Ground Effects
AGARD CP-465, April 1990

Computational Methods for Aerodynamic Design (Inverse) and Optimization
AGARD CP-463, March 1990

Applications of Mesh Generation to Complex 3-D Configurations
AGARD CP-464, March 1990

Fluid Dynamics of Three-Dimensional Turbulent Shear Flows and Transition
AGARD CP-438, April 1989

Validation of Computational Fluid Dynamics
AGARD CP-437, December 1988

Aerodynamic Data Accuracy and Quality: Requirements and Capabilities in Wind Tunnel Testing
AGARD CP-429, July 1988

Aerodynamics of Hypersonic Lifting Vehicles
AGARD CP-428, November 1987

Aerodynamic and Related Hydrodynamic Studies Using Water Facilities
AGARD CP-413, June 1987

Applications of Computational Fluid Dynamics in Aeronautics
AGARD CP-412, November 1986

Store Airframe Aerodynamics
AGARD CP-389, August 1986

Unsteady Aerodynamics — Fundamentals and Applications to Aircraft Dynamics
AGARD CP-386, November 1985

Aerodynamics and Acoustics of Propellers
AGARD CP-366, February 1985

Improvement of Aerodynamic Performance through Boundary Layer Control and High Lift Systems
AGARD CP-365, August 1984

Wind Tunnels and Testing Techniques
AGARD CP-348, February 1984

Aerodynamics of Vortical Type Flows in Three Dimensions
AGARD CP-342, July 1983

Missile Aerodynamics
AGARD CP-336, February 1983

Prediction of Aerodynamic Loads on Rotorcraft
AGARD CP-334, September 1982

Wall Interference in Wind Tunnels
AGARD CP-335, September 1982

Foreward and Introduction

Computational Fluid Dynamic (CFD) methods based on the Euler equations have been a subject of intensive research and development over the past 10 to 15 years. They have now reached a stage where applications are an almost routine matter in most aerospace industries, research laboratories and universities.

After the pioneering work of Lax and Wendroff in the early sixties there was a first, relatively short burst of activities involving Euler methods in the late sixties and early seventies. These activities were closely related to the reviving interest in transonic flow and supercritical wing technology in particular; they were soon to be overtaken by developments in transonic potential flow methods triggered by the work of Murman and Cole.

The main reasons for this shift of attention were an apparent lack of robustness of the Euler methods that had evolved so far but in particular the fact that potential flow methods offered an almost order-of-magnitude reduction of computational effort. Given the level of computer technology at the time, the latter was mandatory if computational transonics were to become a practical tool for the aerospace industry.

While the concentrated efforts on transonic potential flow methods of the seventies and early eighties led to widespread use of such methods in aerospace applications,* a revival of interest in Euler methods was bound to happen and indeed did take place around 1980, in particular through the interest created by the works of Rizzi and Jameson. The main driving factors for this revival were:

— the inherent limitations of potential flow theory, in particular the inability to model vorticity;

— the fact that computer power had increased by a factor 10 to 15 since around 1970;

— the fact that a new generation of more efficient discretization schemes and algorithms held promise for future practical applications.

Research on discretization schemes and algorithms for the Euler equations continues today; not in the least because of the fact that the convective (i.e. "Euler"-) fluxes play a dominant role in CFD methods at the next highest level in mathematical flow modeling, that is, the Navier-Stokes equations. At the same time however, Euler "codes" have reached a level of proliferation in the aerospace aerodynamic community that justifies a review of the state of the art. It is worth noting that Euler methods not only are being used for the simulation of flows for which the modeling of vorticity is mandatory, such as leading-edge vortex flows and flows involving propulsion simulation, but that they are also being used for non-vortical flows around complex geometries, since transonic potential flow methods for complex configurations did not fully mature.

The objective of this AGARDograph, then, is to provide a survey of the state-of-the-art in Computational Aerodynamics Based on the Euler Equations. In terms of technology application, it concentrates on the numerical simulation of external flows about aerospace vehicles.

Internal flows and turbomachinery applications are not extensively treated but touched upon where considered appropriate. In terms of "audience" this AGARDograph is, in the first place, aimed at the applied computational aerodynamicist who wants to get started in this field. However, it might also assist the aerodynamic engineering manager in judging whether his CFD-tools are sufficiently "state-of-the-art" and, if not, in what direction improvement or extension of capabilities shoud be sought. Finally, it might also help the research community to identify niches for further research. For those readers who would like to consult basic text books on CFD and Euler methods, some general references are suggested at the end of this Foreword and Introduction.

Chapter 1 is intended to provide a background of the fluid and thermodynamic theory required for understanding the physics modelled by the Euler equations. Chapter 2 describes numerical schemes and algorithms. Although this is done from a CFD specialist's point of view, the reader should be able to identify the algorithm descriptions given in this chapter with the methods given in the following chapter. In order to meet the objectives for this report, it was felt that the core of the AGARDograph should consist of a survey and description of numerical schemes and algorithms, capabilities, and limitations of the major Euler codes that are currently in use in the NATO countries. For that purpose, requests for information were sent to institutions, industries and individuals who, to the authors' knowledge, were or had been active in this area. The response has been collected in Chapter 3, with examples of application given in Chapter 4.

* A survey can be found in AGARDograph 266 "Applied Computational Transonic Aerodynamics" by T. Holst, et al. 1982.

The preparation of this AGARDograph has been a team effort involving scientists from both Europe and North America. Each of the authors contributed to several if not all of the chapters. In addition, each author took responsibility for coordinating the efforts for the following:

Chapter 1	Nigel Weatherill
Chapter 2	Philippe Morice
Chapter 3	Herbert Rieger and Wolfang Schmidt
Chapter 4	Jim Thomas and Kyle Anderson

Writing it and putting it all together was a substantial amount of work. We thank the authors for their efforts and enthusiasm and their organisations for making it possible.

We also thank all the individuals from the NATO countries that provided material to the authors. Without their help this AGARDograph would have been less complete.

Joop Slooff,
Wolfgang Schmidt
Editors,
Fluid Dynamics Panel

General References

1. Hirsch, Ch. "Numerical Computation of Internal and External Flows", Vols. 1 & 2. John Wiley & Sons Ltd., 1988/90. ISBN 0 471 91762 1/0 471 92351 6.

2. Eberle, A., Rizzi, A., Hirshel, E.H., "Numerical Solutions of the Euler Equations for Steady Flow Problems", Notes on Numerical Fluid Dynamics, Vol. 34 Vieweg, Braunschweig/Wiesbaden, 1991.

Avant-propos et introduction

Les méthodes de l'aérodynamique numérique (CFD) basées sur les équations d'Euler ont fait l'objet de recherches et développement intensifs depuis une quinzaine d'années. Aujourd'hui ces méthodes sont appliquées de façon quasi-courante, dans la plupart des industries aérospatiales, des laboratoires de recherche et des universités.

Les travaux d'avant-garde de Lax et Wendroff du début des années soixante ont été suivis d'une première période d'activités, relativement courte, concernant les méthodes d'Euler, à la fin des années soixante et au début des années soixante-dix. Ces activités étaient étroitement liées au regain d'intérêt qu'il y avait à l'époque pour les écoulements transsoniques et les technologies de profil d'aile supercritique en particulier. Cependant, elles furent rapidement dépassées par le développement des méthodes de calcul des écoulements potentiels transsoniques engendrées par les travaux de Murman et Cole.

Les principales raisons de ce changement de direction furent le manque de robustesse apparent des méthodes d'Euler de l'époque et en particulier le fait que les méthodes de calcul des écoulements potentiels permettaient la réduction du temps de calcul d'environ un ordre de grandeur. Considérant le niveau des technologies de l'informatique à l'époque, cette réduction s'imposait si l'on voulait que l'aérodynamique transsonique numérique puisse devenir un outil pratique pour l'industrie aérospatiale.

Étant donne que les efforts consacrés aux méthodes de calcul des écoulements potentiels transsoniques dans les années soixante-dix et au début des années quatre-vingts ont conduit à la banalisation de ces méthodes dans les applications aérospatiales, un regain d'intérêt pour les méthodes d'Euler était inévitable et ceci s'est produit en 1980, grâce, en particulier, à l'intérêt suscité par les travaux de Rizzi et Jameson. Les principaux éléments moteur de ce renouveau furent :

— les limitations propres à la théorie de l'écoulement potentiel et, en particulier, la non-faisabilité de la modélisation du rotationnel;

— le fait que la puissance de calcul avait augmenté d'un facteur de l'ordre de 10 à 15 depuis 1970;

— le fait qu'une nouvelle génération de méthodes de discrétisation et d'algorithmes plus efficaces semblaient promettre des applications pratiques à l'avenir.

Les travaux de recherche sur les méthodes de discrétisation et les algorithmes pour les équations d'Euler se poursuivent aujourd'hui, principalement à cause du fait que les flux convectifs (c'est-à-dire «d'Euler») jouent un rôle prédominant dans les méthodes de l'aérodynamique numérique au niveau immédiatement supérieur de la modélisation mathématique de l'écoulement, c'est-à-dire au niveau des équations Navier-Stokes. Parallèlement à ces développements pourtant, les «codes» Euler ont atteint un niveau de prolifération au sein de la communauté de l'aérodynamique aérospatiale qui justifie une revue de l'état de l'art.

Il convient de noter, ici, que les méthodes d'Euler ne sont pas utilisées uniquement pour la simulation d'écoulements pour lesquels la modélisation du rotationnel est obligatoire, tels que les écoulements tourbillonnaires de bord d'attaque et les écoulements avec simulation de la propulsion car, en raison du fait que les méthodes de calcul des écoulements potentiels transsoniques ne sont jamais arrivées à maturité, les codes d'Euler sont utilisés aussi pour les écoulements non tourbillonnaires autour de géométries complexes.

L'objectif de cette AGARDographie est, donc, de donner un aperçu de l'état de l'art dans le domaine de l'aérodynamique à partir des équations d'Euler. En matière d'application de la technologie l'accent est mis sur la simulation numérique d'écoulements externes autour de véhicules aérospatiaux. Les flux internes et les applications turbomachines ne sont pas traités en profondeur, mais simplement abordés, le cas échéant.

En matière de «public», cette AGARDographie est destinée, en premier lieu, à l'aérodynamicien concerné par le CFD qui souhaite se familiariser avec ce domaine. Cependant, elle est susceptible d'intéresser aussi le manager d'étude et de conception en matière d'aérodynamique de génie aérodynamique et de lui faire savoir si ses outils CFD correspondent à «l'état de l'art» ou sinon, quelles sont les améliorations et quelles sont les nouvelles capacités à rechercher. Enfin, elle pourrait apporter une aide à la communauté des chercheurs dans l'identification de projets futurs. Pour ceux qui souhaiteraient consulter des ouvrages de caractère général sur l'aérodynamique numérique et les méthodes d'Euler, un certain nombre de références sont proposées à la fin de cette Introduction et avant-propos.

Le chapitre 1 présente l'essentiel de la théorie de la thermodynamique et des fluides nécessaire à la compréhension de la physique modelée par les équations d'Euler. Le chapitre 2 donne la description de méthodes numériques et d'algorithmes. Bien que cette description soit donnée du point de vue du spécialiste CFD, le lecteur devrait pouvoir identifier les descriptions des algorithmes données dans ce chapitre en se servant des méthodes données dans le chapitre suivant. Compte tenu de ces objectifs, les auteurs partageaient l'avis que l'essentiel de l'AGARDographie devait consister en un tour d'horizon et un descriptif des théories numériques et des algorithmes, ainsi que des capacités et des limitations des principaux codes d'Euler actuellement utilisés par les pays membres de l'OTAN. Ayant cet objectif en vue, des demandes ont été adressées aux établissements, aux industries et aux particuliers, qui, à la connaissance des auteurs, étaient, ou avaient été actifs dans ce domaine. Les réponses obtenues sont exposées au chapitre 3, avec des exemples d'applications au chapitre 4.

La rédaction de cette AGARDographie a été un véritable travail d'équipe, mobilisant des scientifiques de l'Europe et de l'Amérique du Nord. Chacun des auteurs a contribué à plusieurs, sinon à l'ensemble des chapitres. En outre, chaque auteur était responsable de la coordination des efforts consacrés à un chapitre en particulier, comme suit :

Chapitre 1	Nigel Weatherill
Chapitre 2	Philippe Morice
Chapitre 3	Herbert Rieger et Wolfang Schmidt
Chapitre 4	Jim Thomas et Kyle Anderson

La rédaction et la mise en forme ont représenté un travail considérable. Nous tenons à remercier les auteurs de leurs efforts et leur enthousiasme, ainsi que leurs organisations respectives, sans lesquelles cette publication n'aurait pas vu le jour.

Nos remerciements sont également dûs aux différentes personnes de la majorité des pays de l'OTAN qui ont fourni des informations aux auteurs. Sans leur concours, cette AGARDographie aurait été moins complète.

Nomenclature

The objective has been to define all symbols locally in the text. The main symbols and notations are summarized below.

General

Bold
vectors of variables indicated in **bold**

matrices indicated in **BOLD** capitals

$\overline{\text{overbar}}$
geometrical/physical vectors indicated by an overbar

$\overline{\textbf{Bold}}$
tensors indicated in $\overline{\textbf{Bold}}$ with overbar

$\hat{\text{circumflex}}$
indicates quantity expressed in generalized coordinates

$\tilde{\text{tilde}}$
denotes quantity obtained through averaging process

Arabic

a	speed of sound
C_D	drag coefficient
C_l	lift coefficient
C_Y	side force coefficient
C_l	rolling moment coefficient
C_p	pressure coefficient $(p-p_\infty)/0.5*\rho_\infty U_\infty^2$
CFL	Courant number
c_p	specific heat at constant pressure
c_v	specific heat at constant volume
$\bar{D}$	dissipation flux tensor
DP/P	$= (p-p_\infty)/p_\infty$
E	total energy per unit volume
e	internal energy per unit mass
$\bar{F}$	Flux vector/tensor with components **f,g,h** (or **F,G,H**)
$f,\bar{f}$	force (vector)
g	gravitational constant
H	total enthalpy
h	enthalpy per unit mass
$\bar{i}, \bar{j}, \bar{k}$	cartesian unit vectors
J	Jacobian of coordinate transformation
k	reduced frequency
M	Mach number
$\bar{n}$	normal vector
P_r	Prandtl number
p	pressure
q	vector of primitive variables
R	universal gas constant, residual
R^*	Riemann variables
R_e	Reynolds number
$r,\bar{r}$	position (vector)
S	discontinuity surface
s	entropy per unit mass
T	temperature
t	time
U,V,W	contra-variant velocity components
$\bar{u}$	velocity of moving surface
V	volume
v	specific volume
$\bar{v}$	velocity vector with cartesian components u,v,w
w	vector of conserved variables
x,y,z	cartesian coordinates

Greek

α	angle of attack
β	angle of side slip
Γ	circulation
γ	ratio of specific heats
Δt	time step
$\Delta x, \Delta y, \Delta z$	spatial steps
$\Delta\xi, \Delta\eta, \Delta\zeta$	
δ	finite difference (operator)
$\zeta, \bar{\zeta}$	vorticity (vector)
θ	weighing factor in generalized explicit/implicit difference scheme (section 2.1)
λ	spectral radius or eigenvalue
μ	averaging operator
ν	kinematic viscosity
ξ	weighting factor in generalized forward/backward difference formulae (section 2.1)
ξ,η,ζ	curvilinear coordinates
ρ	density
Σ	boundary of domain Ω
σ	$= \Delta t/\delta x$
τ	compressibility of a gas or fluid, artificial time
Ψ	stream function
Ω	volume of a domain
ω	angular velocity, relaxation factor

Subscripts

i,j,k	refers to spatial mesh point location
min	minimum value
max	maximum value
m	mean value
n	normal (component)
o	stagnation value
T	at constant temperature
t	tangential component, time derivative
x,y,z	components in x,y,z,/ξ,η,ζ directions
ξ,η,ζ	derivatives with respect to x,y,z/ξ,η,ζ
∞	freestream value

Superscripts

n	iteration/time level
T	transposed (matrix)

Contents

	Page
Recent Publications of the Fluid Dynamics Panel	iii
Foreword and Introduction	v
Nomenclature	ix
CHAPTER 1 — Basic Theory	1
1.1 FLUID, GAS AND THERMODYNAMICS	1
1.1.1 Compressibility of Gases	1
1.1.2 Gas Laws	2
1.1.3 References	4
1.2 CONSERVATION EQUATIONS FOR INVISCID FLOWS	5
1.2.1 Governing Equations	5
1.2.2 Governing Equations in Conservative Integral Form	7
1.2.3 Governing Equations in Conservative Differential Form	8
1.2.4 Rotating Frame of Reference	8
1.2.5 Physics of Inviscid Flows	8
1.2.6 References	9
1.3 JUMP RELATIONS, WEAK SOLUTIONS, RIEMANN PROBLEM	10
1.3.1 Contact Discontinuity-Vortex Sheet	10
1.3.2 Shock Surfaces	10
1.3.3 Weak Solutions and Uniqueness	11
1.3.4 The Riemann Problem	11
1.3.5 References	12
1.4 BOUNDARY CONDITIONS	13
1.4.1 Characteristic Equations	13
1.4.2 Numerical Procedures	15
1.4.3 Inflow/Outflow Boundaries	16
1.4.4 Surface Boundaries	17
1.4.5 Propulsion Simulations	20
1.4.6 References	21
1.5 EXACT SOLUTIONS	22
1.5.1 Similarity	22
1.5.2 Hodograph Solutions	23
1.5.3 Shockless Airfoils	23
1.5.4 Nonunique Solutions	23
1.5.5 Exact Solutions for Supersonic Flow	24
1.5.6 Riemann Problem	24
1.5.7 References	24
CHAPTER 2 — Numerical Schemes and Algorithms	27
2.1 DISCRETIZATION TECHNIQUES	27
2.1.1 Finite Difference Techniques	28
2.1.2 Finite Volume Techniques	30

2.1.3 Finite Element Techniques 31
2.1.4 References 34

2.2 COMPUTATIONAL GRIDS 36
2.2.1 Structured Grids from Partial Differential Equations 37
2.2.2 Structured Grids from Algebraic Methods 40
2.2.3 Structured Grids from Variational Methods 41
2.2.4 The Multiblock Approach 42
2.2.5 Unstructured Grid Methods 44
2.2.6 Other Grid Types 47
2.2.7 Surface Grids 48
2.2.8 Grid Adaptivity Techniques 49
2.2.9 References 51

2.3 SPACE DISCRETIZATION — DISSIPATION 54
2.3.1 Artificial Dissipation Models for Central-Difference Schemes 54
2.3.2 Upwind Differencing 56
2.3.3 Extension to Real Gases 66
2.3.4 References 66

2.4 TIME DISCRETIZATION 69
2.4.1 Explicit Schemes 69
2.4.2 Implicit Schemes 74
2.4.3 References 83

2.5 CONVERGENCE ACCELERATION 86
2.5.1 Local Time Stepping 86
2.5.2 Residual Smoothing 86
2.5.3 Vector Sequence Extrapolation 88
2.5.4 Multigrid Acceleration 91
2.5.5 GMRES 95
2.5.6 Preconditioning 95
2.5.7 Enthalpy Damping 97
2.5.8 References 98

2.6 SPECIAL METHODS 100
2.6.1 Space-marching Techniques 101
2.6.2 Shock-fitting Techniques 103
2.6.3 Inverse Design and Free-surface Flows 104
2.6.4 References 105

CHAPTER 3 — Survey of Major Individual Euler Codes and their Capabilities 107

3.1 CODES FROM NORTH-AMERICA 108
3.1.1 Overview 108
3.1.2 Presentation of Individual Codes 109
3.1.3 Points of Contact 114
3.1.4 References 116

3.2 CODES FROM FRANCE, ITALY AND BELGIUM 118
3.2.1 Overview 118
3.2.2 Presentation of Individual Codes 119
3.2.3 Points of Contact 121
3.2.4 References 122

3.3 CODES FROM GERMANY AND SWITZERLAND 126
3.3.1 Overview 126
3.3.2 Presentation of Individual Codes 129
3.3.3 Points of Contact 131
3.3.4 References 133

3.4 CODES FROM GREAT BRITAIN, THE NETHERLANDS AND SCANDINAVIA 136
3.4.1 Overview 136
3.4.2 Presentation of Individual Codes 137
3.4.3 Points of Contact 139
3.4.4 References 139

CHAPTER 4 — Applications 143

4.1 AIRFOILS 143
4.1.1 NACA 0012 Airfoil 143
4.1.2 Datum Solutions 143
4.1.3 Multi-element Airfoils 145
4.1.4 6:1 Ellipse 145
4.1.5 Heavy-Gas Airfoil Computations 145
4.1.6 References 146
Figures 4.1.1 - 4.1.11 147

4.2 WINGS 155
4.2.1 ONERA M6 Wing 155
4.2.2 DFVLR F4 Wing 155
4.2.3 RAE Wing-Fuselage 155
4.2.4 F5 Wing 156
4.2.5 References 156
Figures 4.2.1 - 4.2.5 157

4.3 VORTICAL FLOWS 161
4.3.1 Delta Wings 161
4.3.2 Delta Wing in Yaw 161
4.3.3 Double-Delta Wing 162
4.3.4 Vortex Breakdown 162
4.3.5 IEPG Vortex Flow Computation/Experiment 162
4.3.6 Wing Canard 163
4.3.7 Asymmetric Cone Flows 163
4.3.8 References 163
Figures 4.3.1 - 4.3.10 165

4.4 AIR-VEHICLE CONFIGURATIONS 173
4.4.1 Subsonic Transport Aircraft 173
4.4.2 Supersonic Transport Aircraft 174
4.4.3 Fighter Aircraft Configurations 175
4.4.4 Missile Configurations 176
4.4.5 Store Configurations 177
4.4.6 Reentry Configurations 177
4.4.7 References 178
Figures 4.4.1 - 4.4.15 181

4.5 PROPULSION SIMULATIONS 196
4.5.1 NACA Propeller 196
4.5.2 Hovering Rotor 196
4.5.3 Turbine and Compressor Cascades 197
4.5.4 Fan Stage Passage 197
4.5.5 Propeller Slipstream 198
4.5.6 Turbofan Nacelle Analysis System 198
4.5.7 Inlets 198
4.5.8 Scramjets 199
4.5.9 Nozzles 199
4.5.10 References 199
Figures 4.5.1 - 4.5.9 201

4.6 HYPERSONIC FLOWS 208
4.6.1 Blunt Cylinders 208
4.6.2 Shock-Shock Interactions 208
4.6.3 Double Ellipsoid 208
4.6.4 Hermes 209
4.6.5 References 209
Figures 4.6.1 - 4.6.6 211

4.7 UNSTEADY FLOWS 217
4.7.1 Forced Oscillations 217
4.7.2 Flutter Predictions 218
4.7.3 References 218
Figures 4.7.1 - 4.7.5 219

4.8 DESIGN APPLICATIONS 224
4.8.1 Airfoils 224
4.8.2 Engine Nacelle 224
4.8.3 Supersonic Conical Wings 225
4.8.4 Low-Drag Bodies 225
4.8.5 References 226
Figures 4.8.1 - 4.8.5 227
COLOUR PLATES 4.1 - 4.12 233

Concluding Remarks 237

Chapter 1

Basic Theory

1.1 FLUID, GAS AND THERMO DYNAMICS

A general definition of a fluid is any material that cannot sustain a tangential, or shearing force when at rest and that undergoes a continuous change in shape when subjected to a stress. This continuous and irrecoverable change of position of one part of the material relative to another part when under shear stress constitutes flow, a characteristic property of fluids. In contrast, the shearing forces in a solid, held in a twisted or flexed position, are maintained; the solid undergoes no flow and can spring back to its original shape.

Various simplifications, or models, of fluids have been devised to analyze fluid flow. The simplest model, and the one primarily of interest in this document, called a perfect, or ideal fluid is one that is unable to conduct heat or to offer any internal resistance to one portion flowing over another. A perfect fluid cannot sustain a tangential force, that is, it lacks viscosity and is called an inviscid fluid.

The study of the effects of forces and energy on liquids and gases is known as fluid mechanics. Like other branches of classical mechanics, the subject subdivides into statics (often called hydrostatics) and dynamics (fluid dynamics, hydrodynamics, or aerodynamics). Hydrostatics is a comparatively elementary subject with a few classical results of importance but little scope for further development. Fluid dynamics, in contrast, is a highly developed branch of science that has been the subject of continuous and expanding research activity from around 1800 to the present day.

The development of fluid dynamics has been strongly influenced by its numerous applications. In the area of aeronautical engineering and the study of flight the importance of fluid dynamics is obvious.

Traditionally, fluid dynamics has been studied both theoretically and experimentally. The phenomena of fluid motion, as will be described in the following sections, are governed by known laws of physics - conservation of mass, the laws of classical mechanics (Newton's laws of motion), and the laws of thermodynamics. As will be demonstrated, these can be formulated as a set of nonlinear partial differential equations, and in principle one might hope to infer all the phenomena from these. In practice, this has not been possible; the mathematical theory is difficult and the nonlinear nature of the equations are not amenable to classical mathematical approaches. More recently, with the advent of high speed computers, the science of computational fluid dynamics has emerged which aims to solve the governing fluid flow equations with the use of numerical techniques which are carried out in the computer. However, the complexity of the problems associated with either the mathematical or computational approaches to fluid dynamics necessitates the continuing research in observations of fluid motion both in the laboratory and in nature.

Traditionally, liquids and gases are classified together as fluids because, over a wide range of situations, they have identical equations of motion and thus exhibit the same phenomena. However, in the applications to be discussed here, flow speeds are comparable with that of the speed of sound, where the density of the fluid changes significantly. This phenomena is of practical importance only for gases. However, throughout the document the term fluid will be assumed to be used in a generic sense, with the implied assumption of application to gases.

Some of the issues alluded to in this introduction will now be considered in further detail. The intention is that, within this chapter, the basic formulations required in the study of computational inviscid aerodynamics will be presented.

1.1.1 Compressibility of Gases

In general terms, a compressible flow is one in which there is a variation in density of the fluid. This rather vague definition must be enhanced if it is to be of value. Consider a small element of fluid of volume V, in which the pressure exerted upon it is p. If the pressure is increased by a small amount δp, the volume of the element will be compressed by an amount δV. Hence the compressibility of the fluid τ, can be defined as

$$\tau = -\frac{1}{V}\frac{\delta V}{\delta p},$$

which, in the limit, can be expressed in derivative form as,

$$\tau = -\frac{1}{V}\frac{dV}{dp} \qquad (1.1.1)$$

The negative sign indicates a decrease in volume for an increase in pressure.

Compressibility is, therefore, the fractional change in volume of the fluid per unit change in pressure. However, this description is not adequate, since when a gas is compressed the temperature will, in general, change. Hence, it is necessary to introduce the idea of an *isothermal compressibility* in which the temperature is held constant. The definition is now extended to

$$\tau_T = -\frac{1}{V}\left(\frac{dV}{dp}\right)_T \qquad (1.1.2)$$

where the subscript T denotes that the change in volume takes place at constant temperature.

Alternatively, if no heat is added to, or taken away from, the fluid element (i.e. the compression is adiabatic) then the compressibility is *isentropic* and is defined as

$$\tau_S = -\frac{1}{V}\left(\frac{dV}{dp}\right)_S \qquad (1.1.3)$$

where now the subscript S denotes that the change in volume takes place at constant entropy. If the fluid element is assumed to have a unit mass, v is then the specific volume and the density $\rho = \frac{1}{v}$. Eq. (1.1.1) can then be expressed in terms of the density in the form

$$\tau = \frac{1}{\rho}\frac{d\rho}{dp} \qquad (1.1.4)$$

It is evident from Eq. (1.1.4) that whenever a fluid experiences a change in pressure, dp, the corresponding change in density will be $d\rho$, where

$$d\rho = \rho\,\tau\,dp \qquad (1.1.5)$$

From this statement, it is clear that all fluids are compressible to some extent. However, some fluids have very low values of

compressibility. For water, $\tau_T = 5 \times 10^{-10}$ m^2/N at 1 atmosphere(atm), whereas, gases have a high compressibility, typically, for air $\tau_T = 5 \times 10^{-5}$ m^2/N at 1 atm. The concept of an incompressible fluid is an assumption which is only true if density changes are negligibly small. Primarily, in this document flow conditions are such that changes in density are significant and the approximation of an incompressible fluid cannot be made.

1.1.2 Gas Laws

The relationships between the volume of a gas and its temperature and pressure are fundamental properties of gases. These relationships are described in the so-called gas laws.

The first law is due to Boyle and states that the volume of a given quantity of gas, V, varies inversely with the pressure exerted on it, if the temperature is constant. From experiment the constant of proportionality can be found and then Boyle's law can be written

$$V=k_1/p.$$

The second law is due to Charles and states that the volume of gas varies directly as its absolute temperature, if the pressure is kept constant. Again the constant of proportionality can be determined from experiment and then Charles's law written as

$$V=k_2T.$$

If both pressure and temperature are changed at the same time, both these equations must be used to calculate the change in volume, pressure or temperature. A single equation can be derived from Boyles's and Charles's laws: the product of the pressure and the volume equals the product of the temperature and a constant, called the universal gas constant, R. This can be written as

$$pV' = RT, \qquad (1.1.6a)$$

where V' is the volume of one mole of gas.

It is possible to also derive this fundamental gas relationship from the kinetic theory of gases. It is known that molecules possess a force field which interacts with neighbour molecules. A perfect gas is one in which these intermolecular forces are negligible. From this assumption it is possible to derive from kinetic theory the equation of state for a perfect gas. Historically, the equation of state was derived in the form

$$pV = MRT \qquad (1.1.6b)$$

where V is the volume of the system, M is the mass of the system, and is equivalent to (1.1.6a). Many different forms of this equation can be derived. It follows that, on dividing by the mass of the system, (1.1.6) can be expressed as

$$pv = RT \qquad (1.1.7)$$

where v is the specific volume, or, alternatively, using the fact that $\rho = \frac{1}{v}$, as

$$p = \rho RT . \qquad (1.1.8)$$

Other expressions are also possible.

It is worth considering the accuracy of the assumption of a perfect gas. Experimentally, it has been determined that, at low pressures (near 1 atm or less) and at high temperatures (273 Kelvin(K) and above), the value pv/RT for most pure gases deviates from unity by less than 1 percent.

An ideal gas, or perfect gas obeys the gas relation.

However, at very low temperatures and high pressures, molecules are more closely packed and intermolecular effects can be significant. Under these conditions, the gas is defined as a real gas. In such cases the perfect gas relations (1.1.6 - 1.1.8) are replaced by more accurate relations.

The first serious attempt to understand the behaviour of real gases was made by van der Waals in 1873. The van der Waals equation,

$$\left(p + \frac{a}{v^2} \right) (v - b) = RT \qquad (1.1.9)$$

contains the two adjustable parameters a and b which are dependent upon the gas. Eq. (1.1.9) shows that, with molecular attraction, the pressure p is incremented by the term a/v^2, whilst the finite volume occupied by the molecules reduces from v to (v-b). There are now over 100 emperical thermal equations of state available for a whole range of gases. Of the two best known, one contains 5 adjustable parameters whilst the other contains 8 such parameters. These equations can reproduce the complete range of pressure, volume and temperature behaviour from gas phase to condensation. The main use of such relations is that, through interpolation, the behaviour of the gas at non-measured values can be approximated.

One of the most convenient ways of expressing the thermal equation of state for a real gas is to use the virial equation of state. This can be expressed as

$$\frac{pv}{RT} = 1 + \frac{B(T)}{v} + \frac{C(T)}{v^2} + \frac{D(T)}{v^3} + \ldots$$

The quantities B(T), C(T), D(T), etc. are called the second, third and fourth virial coefficients, the first virial coefficient being unity. They are all independent of the gas pressure and density, and dependent upon the temperature only. Clearly for a perfect gas this reduces to the ideal gas law.

In the microscopic view of a gas, individual molecules are in random motion, colliding with other molecules. Evidently, there are many forms of energy inherent to motion of this type. The internal energy, e, of a gas is the total sum of all these different energies. If the particles of the gas are in a state of maximum disorder, then the system of particles is in equilibrium, i.e. no gradients in velocity, pressure, temperature and chemical concentrations exist in the system. It proves appropriate to introduce a further property of a gas and that is enthalpy. The enthalpy, h, is defined, per unit mass, as

$$h = e + pv \qquad (1.1.10)$$

If a gas is not chemically reacting, and intermolecular forces are ignored, then the resulting system is a thermally perfect gas where internal energy and enthalpy are functions of temperature only and where the specific heats at constant volume and pressure, c_v and c_p, are also functions of temperature only:

$$e = e(T),\ h = h(T),\ de = c_v\, dT,\ dh = c_p\, dT. \qquad (1.1.11)$$

The ratio $c_p/c_v = \gamma$, where, for air at standard conditions, γ=1.4.

For a real gas it should be noted that the internal energy and enthalpy are functions of both temperature and volume, i.e.

$$e = e(T,v),\ h = h(T,p). \qquad (1.1.12)$$

If the specific heats are constant, the system is a calorically perfect gas, where

$$e = c_v\, T,\ h = c_p\, T. \qquad (1.1.13)$$

Useful expressions for the specific heats are

$$c_p = \frac{\gamma R}{\gamma - 1} \qquad (1.1.14)$$

and

$$c_v = \frac{R}{\gamma - 1} \qquad (1.1.15)$$

Eqs (1.1.14) and (1.1.15) are valid for a thermally or calorically perfect gas but are not valid for either a chemically reacting or a real gas.

In many compressible flow applications the pressures and temperatures are such that the gas can be considered as calorically perfect. However, it should be noted that in some applications associated with hypersonic flows, where temperatures can be high, the assumption of a calorically perfect gas is invalid. At excessive temperatures, where molecules start to dissociate, chemical reactions occur and then the internal energy depends on both the temperature and volume, and the enthalpy on temperature and pressure. As the temperature of a gas increases it changes behaviour from calorically perfect to thermally perfect.

The behaviour of gases at high temperatures can be predicted accurately using statistical mechanics. Many varied forms of behaviour can be studied using these ideas and the fundamental relationships relating temperature and pressure to density and enthalpy can be derived.

Air at normal room temperature and pressure is composed of approximately 79% nitrogen, 20% oxygen and 1% trace species. Hence, to a reasonable level of approximation the behaviour of the 2 species, oxygen and nitrogen are important. As the temperature is increased to within the range 2500K<T<9000K, chemical reactions take place between oxygen and nitrogen, producing not only O_2 and N_2 but also O, N, NO, NO^+, and e^-. If a fixed temperature and pressure are maintained, then in time the condition of chemical equilibrium is reached. In the case of an equilibrium chemically reacting gas, the chemical composition i.e. the amounts of each species, is determined uniquely by the pressure and temperature. In the time required to reach steady state, the behaviour is that of a non-equilibrium chemically reacting gas.

Details of the gas laws for these different states can be found in appropriate texts[1,2]. For an equilibrium chemically reacting gas, in addition to the flow equations, it is necessary to determine the chemistry of the gas. In general, if the gas has K species and H elements then (K-H) independent chemical equations are needed together with equations for mass balance and partial pressures. For air, with elements O, N and electric charge e^-, and 7 species, O_2, N_2, O, N, NO, NO^+, and e^- this results in 7 non-linear, simultaneous algebraic equations.

When non-equilibrium chemical reactions are considered, it is necessary to also determine the evolution of the chemical species. This is achieved by solving the appropriate chemical rate equations, which take the form of

$$\frac{\partial \rho \alpha_i}{\partial t} + \frac{\partial \rho \alpha_i u}{\partial x} + \frac{\partial \rho \alpha_i v}{\partial y} + \frac{\partial \rho \alpha_i w}{\partial z} = Q_i$$

where α_i are the mass fractions of the chemical species, and Q_i are the source terms.

It is useful at this stage to consider further definitions related to compressible gas dynamics. The first law of thermodynamics states that the heat added to a system, δq, and the work done on the system, δw, cause a change in energy, and since the system is stationary, this change in energy, δe, is

$$\delta e = \delta q + \delta w$$

For a given δe, there are many ways in which heat can be added and work done on the system. Some of particular interest are

Adiabatic process - one in which no heat is added to or taken from the system.

Reversible process - one in which no dissipative phenomena occur, i.e.. where the effects of viscosity, thermal conductivity and mass diffusion are absent.

Isentropic process - one which is both adiabatic and reversible.

The second law of thermodynamics states in which direction a thermodynamic process will occur. To provide a more formal definition, the state variable entropy is introduced as

$$ds = \frac{\delta q}{T},$$

where s is the entropy of the system, δq is an incremental amount of heat added reversibly to the system, and T is the system temperature. Alternative definitions exist, such as

$$ds = \frac{\delta q}{T} + ds_{irrev} \qquad (1.1.16)$$

which clearly states that the change in entropy during an incremental process is equal to the actual heat added divided by the temperature plus a contribution from the irreversible dissipative phenomena of viscosity, thermal conductivity and mass diffusion occurring within the system. These dissipative phenomena always increase the entropy

$$ds_{irrev} > 0.$$

The entropy can be computed from a variety of expressions, such as

$$T ds = de + p\, dv \qquad (1.1.17)$$

and

$$T ds = dh - v\, dp \qquad (1.1.18)$$

Above, an isentropic process was defined as adiabatic and reversible and hence ds = 0. i.e. the entropy is constant.

Disturbances in fluids are transmitted through the flow field by molecular action. Molecules collide with their neighbours, transferring the newly acquired energy to others. This wave of energy travels through the air at a velocity that is related to the molecular velocity. The energy increase causes the pressure, as well as density and temperature, to change slightly. In this way, any disturbance is propagated throughout the flow field. As will be described, the velocity at which disturbances are propagated is of central importance to the field of aerodynamics.

Consider a sound wave moving with velocity, a, through a gas. The flow through the sound wave is one dimensional. Applying conservation of mass across the wave front, leads to

$$\rho a = (\rho + d\rho)(a + da), \qquad (1.1.19)$$

which, if products of small quantities are ignored, leads to

$$a = -\rho \frac{da}{d\rho} \quad . \qquad (1.1.20)$$

Conservation of momentum across the wave front, can be similarly be expressed as

$$p + \rho a^2 = (p + dp) + (\rho + d\rho)(a + da)^2 \qquad (1.1.21)$$

Expanding and using Eq. (1.1.20) to find da, gives an expression for the speed of sound;

$$a^2 = \frac{dp}{d\rho} \qquad (1.1.22)$$

The changes which occur across the wave front are small. This implies that the irreversible, dissipative effects of friction and thermal conduction are negligible. Further, there is no heat addition to the system, and hence the process inside a sound wave is isentropic. Hence, the rate of change of pressure with density is an isentropic change and hence Eq. (1.1.22) can be written

$$a^2 = \left(\frac{dp}{d\rho}\right)_S \qquad (1.1.23)$$

This equation demonstrates that the speed of sound is a direct measure of the compressibility of a gas. Using the expression for compressibility, Eq. (1.1.23) can be written as

$$a = \sqrt{\left(\frac{dp}{d\rho}\right)_S} = \sqrt{\frac{\nu}{\tau_S}} \qquad (1.1.24)$$

In turn, for a calorically perfect gas this can be expressed as

$$a = \sqrt{\left(\frac{\gamma p}{\rho}\right)} = \sqrt{\gamma RT} \qquad (1.1.25)$$

It is noted that for a perfect gas the speed of sound is a function of temperature only. At standard sea level conditions the speed of sound is

$$a_S = 340.9 \text{ m/s} = 1117 \text{ ft/s}.$$

For aeronautical flows, a useful classification of flows arises if the local speed, q, is normalised with respect to the speed of sound. This normalisation introduces the concept of the Mach number M, defined as

$$M = \frac{q}{a} . \qquad (1.1.26)$$

Three natural classifications of different flow regimes then follow;

$M < 1$	subsonic flow
$M = 1$	sonic flow
$M > 1$	supersonic flow.

The Mach number plays an important role in aeronautics, since the physical nature of flows is radically different dependent upon the Mach number. Clearly, interesting phenomena are to be expected if an object is travelling at a speed which is greater than that at which disturbances created by it are propagated. Further discussions on this will be given later in this document.

1.1.3 References

1. LIEPMANN H. W. and ROSHKO A., *Elements of Gasdynamics*, Wiley, New York, 1957.

2. ANDERSON J. D. *Modern Compressible Flow*, McGraw-Hill, 1989.

1.2 CONSERVATION EQUATIONS FOR INVISCID FLOWS

The fundamental equations of fluid dynamics have been known for over a century. The French engineer Claude-Louis-Marie Navier and the British physicist George Gabriel Stokes are credited with the original derivation, and hence the governing fluid flow equations are known by their respective names. On the basis of a molecular hypothesis, Navier arrived in 1827 at a theory of elasticity of isotropic solids (solids in which elasticity is uniform in all directions) that contained only one elastic constant. Later in 1845, Stokes, using phenomenological concepts, produced the modern theory which invokes concepts of shear and bulk modulus. Careful experiments later confirmed the work of these two scientists. These equations govern the dynamics of all fluid flow and it is only the imposed boundary conditions and the physical nature of the fluid which distinguishes the different fluid motions.

Before Navier and Stokes, the Swiss scientist Leonhard Euler had derived in 1775 a set of equations valid for a fluid assumed to be non-viscous and non-heat conducting. These equations are a subset of the Navier-Stokes equations where the viscous and heat conduction terms are neglected.

The dynamics of fluid motion are governed by fundamental physical principles. To construct the relevant equations for fluid motion it is necessary to select the appropriate laws of physics and to apply them to a suitable model of the fluid. From these the governing mathematical equations can be extracted. The properties of a fluid will be fully defined once the velocity field $\bar{v}$, pressure p, density ρ and temperature T are known, in the Eulerian sense, as functions of the space coordinate $\bar{r}$ and the time t so as to satisfy a sufficient set of boundary and initial conditions.

All fluids satisfy the laws that mass, momentum and energy are conserved. The conservation of momentum is equivalent to Newton's laws of motion, in particular, the force applied to a body is equal to the product of the body's mass and its acceleration. These physical laws, in principle, are sufficient to enable the equations for fluid motion to be derived. However, it is clearly necessary, before these laws can be implemented, to define and describe how a fluid is to be modelled.

A fluid can be thought of in a number of different ways. For example, on a microscopic scale, a fluid element of volume dV which, although infinitesimal on a macroscopic scale, will contain a representative number of molecules, ndV. For a monatomic gas, it can be assumed that these molecules move at constant but widely different and independent velocities, with collisions between molecules taking place in a random manner. Application of the fundamental physical laws to this model of a fluid, often called the kinetic theory, would result in an appropriate mathematical description of fluid flow.

Alternatively, a fluid can be thought of at a macroscopic level, and the physical laws applied to a closed small fluid element, which is, however, large enough to contain a very large number of molecules so that it can be viewed as a continuous medium. This approach invokes the continuum hypothesis on which many classic theories of fluid motion are based.

Whichever model of a fluid is chosen it is to be expected that the resulting governing equations are equivalent. In this section the equations which govern inviscid, non-conducting fluids will be derived.

1.2.1 Governing Equations

Derivation from Kinetic Theory

The approach relies upon the fact that the Euler equations of fluid mechanics can be obtained by taking moments of the Boltzmann equation, with a Maxwellian velocity distribution function .

The Boltzmann equation from the Kinetic Theory of gases can be expressed as,

$$\frac{\partial f_v}{\partial t} + \bar{v} \cdot \frac{\partial f}{\partial \bar{x}} = J, \qquad (1.2.1)$$

where f_v is the velocity distribution function which gives the number density, in phase space, of molecules with position $\bar{x}$ and velocity $\bar{v}$ at time t. The function, J, on the right hand side of the equation, represents a molecular collision term which vanishes in the Euler limit. The Maxwellian velocity distribution in 2 dimensions is given by

$$f_M = \frac{\rho\beta}{I_o \pi} \exp\left[-\beta(v_1-u_1)^2 - \beta(v_2-u_2)^2 \right] \cdot \frac{1}{I_o}$$

where $\beta = 1/(2RT)$, ρ=mass density, T the temperature, R the gas constant per unit mass, $\bar{v} = (v_1, v_2)^T$ is the molecular velocity vector, $\bar{u} = (u,v)^T$ is the fluid velocity vector, I the internal energy variable corresponding to non-translational degrees of freedom needed to force the given value of γ for the gas consisting of pseudo particles, I_o is the internal energy due to non-translational degrees of freedom , and γ the ratio of specific heats.

The Euler equations can be derived by taking moments of the Boltzmann equation. The appropriate moment vector is defined as

$$\Psi = [1, v_1, v_2, I + (v_1^2+v_2^2)/2]^T \qquad (1.2.2)$$

Applying the moment vector Eq. (1.2.2) to the Boltzmann equation, Eq. (1.2.1), leads to the Euler equations in the familiar differential form,

$$\frac{\partial \mathbf{w}}{\partial t} + \frac{\partial \mathbf{f}}{\partial x_1} + \frac{\partial \mathbf{g}}{\partial x_2} = 0, \qquad (1.2.3)$$

where

$$\mathbf{w} = \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ \rho E \end{bmatrix} \quad \mathbf{f} = \begin{bmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ u(p+\rho E) \end{bmatrix} \quad \mathbf{g} = \begin{bmatrix} \rho v \\ \rho u v \\ \rho v^2 + p \\ v(p+\rho E) \end{bmatrix} \qquad (1.2.4)$$

and the equations of state, with ρ the density, u,v the Cartesian velocity components, p the pressure and E the total energy per unit volume. The relationship between the Boltzmann and Euler equations can be written as

$$\left\langle \Psi, \frac{\partial f_M}{\partial t} + v_1 \frac{\partial f_M}{\partial x_1} + v_2 \frac{\partial f_M}{\partial x_2} \right\rangle = \frac{\partial \mathbf{w}}{\partial t} + \frac{\partial \mathbf{f}}{\partial x_1} + \frac{\partial \mathbf{g}}{\partial x_2} \qquad (1.2.5)$$

where the moments are defined as

$$\langle \Psi, f_M \rangle = \int_0^\infty dI \int_{-\infty}^{\infty} dv_1 \int_{-\infty}^{\infty} dv_2 \, \Psi \, f_M, \qquad (1.2.6)$$

with v_1 and v_2 the components of the molecular velocity vector and f_M the Maxwellian distribution in 2 dimensions[1]. It follows, therefore, that the following relations hold

$$w=<\Psi, f_M>,\ f=<\Psi, v_1 f_M>,\ \text{and}\ g=<\Psi, v_2 f_M> \qquad (1.2.7)$$

The formulation presented thus involves two levels; the Boltzmann level, Eq. (1.2.1), and the Euler level, Eq. (1.2.3), with the connection given by the moments defined in Eq. (1.2.6).

Eulerian Approach to the Governing Equations

From a macroscopic continuum approach the governing equations of fluid motion are derived by considering the three fundamental physical principles that

a) mass is conserved,
b) force = mass x acceleration
c) energy is conserved.

If each of these physical laws are applied in a control volume formulation the governing motion of a fluid can be derived[2].

Consider an arbitrary, but stationary, control volume Ω, as shown in Fig. 1.2.1, bounded by a closed surface Σ, which has an outward unit normal vector $\bar{n}$ at a point A on the surface. Let $d\Sigma$ be an incremental area on the bounding surface around point A. Define $\overline{d\Sigma} = \bar{n}\, d\Sigma$. Let $\bar{v}$ and ρ be the local velocity and density at the point A, respectively, with $\bar{v}$ at an angle θ to $\bar{n}$.

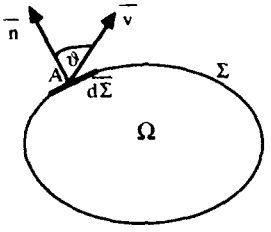


Figure 1.2.1 Domain Ω enclosed by contour Σ.

Mass Conservation

The mass flow through any surface arbitrarily oriented in a flowing fluid is equal to the product of density, the component of velocity normal to the surface, and the area. Thus,

$$\text{mass flow} = \rho\,(V\cos\theta)\,d\Sigma = \rho\, V_n\, d\Sigma = \rho\, \bar{v} \cdot \bar{n}\, d\Sigma$$

The net mass flux into the control volume Ω, through the entire control surface Σ, is the sum of all incremental mass flows, i.e.

$$-\int_{\Sigma} \rho\, \bar{v} \cdot \bar{n}\, d\Sigma$$

The negative sign indicates that the mass flow is into the control volume in the opposite sense of the outward vector $\bar{n}$. The total mass inside an incremental control volume is $\rho\, d\Omega$. Hence, the total mass in Ω is the volume integral

$$\int_{\Omega} \rho\, d\Omega$$

Since, in the absence of any mass sources or sinks, the mass of the fluid is conserved then, the time rate of change of the mass in the domain Ω must equal the change in mass across the domain boundary Σ. Hence, in integral form, this can be written as

$$\frac{d}{dt}\int_{\Omega} \rho\, d\Omega = -\int_{\Sigma} \rho\, \bar{v} \cdot \bar{n}\, d\Sigma \qquad (1.2.8)$$

This is the continuity equation, or conservation of mass equation, in integral form.

Conservation of momenta

Newton's second law can be stated as the time rate of change of momentum of a body equals the net force exerted on it. This principle is valid for the control volume, Ω and when applied leads to the equation for the conservation of momenta.

Consider first the contribution to the forces. The net forces acting on the volume, excluding frictional forces can be thought of as taking two forms. Firstly, body forces acting on the fluid inside Ω. Such forces arise from, for example, electromagnetic or gravitation effects. If $\bar{f}$ denotes the body force per unit mass of fluid, then the force on an incremental volume $d\Omega$ is equal to the product of the mass and the force per unit mass, namely $(\rho d\Omega)\,\bar{f}$. For the entire control volume this is summed to give

$$\int_{\Omega} \rho\, \bar{f}\, d\Omega$$

The second type of force on the fluid arises from surface effects. These arise from pressure and shear stress distributions over the surface. Since, here only inviscid fluids are to be considered, these latter contributions will be ignored. The pressure force acting on an elemental area $\bar{n}\, d\Sigma$ is $-p\,\bar{n}\, d\Sigma$, where the negative sign indicates that the pressure acts inwards. The pressure effect can be summed to give a contribution to the forces from pressure over the complete surface in the form

$$-\int_{\Sigma} p\, \bar{n}\, d\Sigma.$$

The principle of conservation of momenta can be applied within a control volume. The mass, in the control volume, can be expressed as $\rho\, \bar{v} \cdot \bar{n}\, d\Sigma$, which has a momentum of $(\rho\, \bar{v} \cdot \bar{n}\, d\Sigma)\, \bar{v}$. The net rate of flow of momentum is, therefore,

$$-\int_{\Sigma} (\rho\, \bar{v} \cdot \bar{n}\, d\Sigma)\, \bar{v}\ .$$

In addition to this contribution, there can arise a contribution of momentum of $(\rho d\Omega)\, \bar{v}$ for an elemental volume $d\Omega$, from unsteady effects taking place within the control volume Ω. Hence, another contribution should be added in the form of

$$\frac{d}{dt}\int_{\Omega} (\rho\, \bar{v}\, d\Omega)\ .$$

The conservation of momentum now gives

$$\frac{d}{dt}\int_{\Omega} (\rho\, \bar{v})\, d\Omega = -\int_{\Sigma} (\rho\, \bar{v} \cdot \bar{n}\, d\Sigma)\, \bar{v} + \int_{\Omega} \rho\, \bar{f}\, d\Omega - \int_{\Sigma} p\, \bar{n}\, d\Sigma \qquad (1.2.9)$$

Equation (1.2.9) is called the momentum equation for inviscid flow and is presented here in its integral form.

Conservation of Energy

Finally, it is necessary to consider the equation for energy. Energy cannot be created or destroyed, it can only change its form. If the energy principles are applied to the control volume Ω then clearly, the rate of heat added to the fluid plus the rate of work done on the fluid is equal to the rate of change of energy of the fluid as it flows through the control volume i.e. energy is conserved.

If the rate of heat added per unit mass is q, then the rate of heat added to an incremental volume with mass $(\rho d\Omega)$ is $q(\rho d\Omega)$. Summing all such effects with the control volume Ω, gives

$$\int_{\Omega}(q\rho)d\Omega \ .$$

The pressure force acting on the elemental area $d\Sigma$ is $-pd\Sigma$. The rate of work done on the fluid which passes through $\bar{n}\,d\Sigma$ with velocity $\bar{v}$ is therefore, $(-p\,\bar{n}\,d\Sigma)\,\bar{v}$. Hence, over the complete surface the contribution is

$$-\int_{\Sigma} p\,\bar{n}\,.\,v\,d\Sigma$$

The rate of work done by the body force, $\bar{f}$ per unit mass on the elemental volume $d\Omega$ is $(\rho\,\bar{f}\,d\Omega).\,\bar{v}$ and thus the contribution for the entire control volume is

$$\int_{\Omega}(\rho\,\bar{f}\,d\Omega).\,\bar{v}$$

The internal energy, for a stationary fluid was previously represented as e. If the fluid is also in motion then, in addition, it possesses kinetic energy in the form of $\frac{1}{2}\bar{v}^2$. Hence, the total energy per unit mass of the moving fluid is the sum of the internal and kinetic energies, $e+\frac{1}{2}\bar{v}^2$. For an elemental control surface $d\Sigma$, the flux of energy across $d\Sigma$ is $\bar{v}\,.\,\bar{n}\,d\Sigma(e+\frac{1}{2}\bar{v}^2)$. For the complete bounding surface this sums to

$$\int_{\Sigma}\rho\,\bar{v}\,.\,\bar{n}\,(e+\tfrac{1}{2}\bar{v}^2)\,d\Sigma$$

The time rate of change of energy in Ω due to transient variations of the flowfield variables can be expressed as

$$\frac{d}{dt}\int_{\Omega}\left(\rho(e+\tfrac{1}{2}\bar{v}^2)\right)d\Omega.$$

Hence, rearranging the integrands, and applying the law that the time rate of change of the conserved quantity must be equal to the flux through the boundary surface, together with production terms, leads to

$$\frac{d}{dt}\int_{\Omega}\left(\rho(e+\tfrac{1}{2}\bar{v}^2)\right)d\Omega+\int_{\Sigma}\rho\,\bar{v}\,.\,\bar{n}\,(e+\tfrac{1}{2}\bar{v}^2)d\Sigma=$$

$$\int_{\Omega}(q\rho)d\Omega-\int_{\Sigma}p\,\bar{n}\,.\,\bar{v}\,d\Sigma+\int_{\Omega}(\rho\,\bar{f}\,d\Omega).\,\bar{v} \qquad (1.2.10)$$

It should be noted that this energy equation does not include work done by viscous stresses or heat added to the system due to thermal conduction and diffusion. Equation (1.2.10) represents the conservation of energy written in integral form for an inviscid fluid.

The conservation equations derived above Eq. (1.2.8), (1.2.9), and (1.2.10) represent, in 3 dimensions, 5 equations for the 6 unknowns ρ, $\bar{v}=(u,v,w)$, e and T. The system of equations is closed with the addition of an equation of state, namely,

$$p=\rho RT$$

These together with the thermodynamic relation $e=e(T,v)$ which, as discussed previously, simplifies to $e=c_vT$ for a perfect gas, are sufficient equations to analyse inviscid compressible flows of an equilibrium gas. These equations can be written in many different forms, and two are worthy of note here since they will be used later.

1.2.2 Governing Equations in Conservative Integral Form

To standardise the equations a unified representation often used in computational aerodynamics is

$$\frac{\partial}{\partial t}\int_{\Omega}\mathbf{w}d\Omega+\int_{\Sigma}\bar{F}\,.\,\bar{n}\,d\Sigma=\int_{\Omega}\rho\mathbf{f}_e d\Omega \qquad (1.2.11)$$

The conserved variable $\mathbf{w}$ and the Cartesian flux function $\bar{F}$ are given by :

$$\mathbf{w}=\begin{bmatrix}\rho\\ \rho u\\ \rho v\\ \rho w\\ \rho E\end{bmatrix}\quad \mathbf{f}=\begin{bmatrix}\rho u\\ \rho u^2+p\\ \rho uv\\ \rho uw\\ \rho uH\end{bmatrix}\quad \mathbf{g}=\begin{bmatrix}\rho v\\ \rho uv\\ \rho v^2+p\\ \rho vw\\ \rho vH\end{bmatrix}\quad \mathbf{h}=\begin{bmatrix}\rho w\\ \rho uw\\ \rho vw\\ \rho w^2+p\\ \rho wH\end{bmatrix}$$

and the force vector

$$\mathbf{f}=\begin{bmatrix}0\\ f_x\\ f_y\\ f_z\\ \bar{f}\,.\,\bar{v}\end{bmatrix}$$

where, u,v and w are the Cartesian velocity components, E and $H=h+V^2/2$ are, respectively, the total energy and total enthalpy per unit volume. The temperature T is obtained from the equation of state which closes the system, namely

$$p=\rho RT.$$

1.2.3 Governing Equations in Conservative Differential Form

Integral forms of the equations, derived from an analysis of conservation laws within a domain, can be converted to representation at a point using the well known vector identities,

$$\int_{\Sigma} \bar{A} \cdot \bar{n}\, d\Sigma = \int_{\Omega} (\nabla \cdot \bar{A})\, d\Omega \qquad (1.2.12)$$

and

$$\int_{\Sigma} \Theta . d\Sigma = \int_{\Omega} (\nabla \Theta)\, d\Omega \qquad (1.2.13)$$

where $\bar{A}$ and Θ are arbitrary vectors and scalars, respectively.

It can be readily shown that the application of Eq. (1.2.12) and (1.2.13) to Eq.(1.2.11) leads to the following differential forms of the governing equations of inviscid fluid motion.

$$\begin{aligned}
&\frac{\partial \rho}{\partial t} + \nabla.(\rho \bar{v}) = 0 \\
&\frac{\partial(\rho u)}{\partial t} + \nabla.(\rho u \bar{v}) = -\frac{\partial p}{\partial x} + \rho f_x \\
&\frac{\partial(\rho v)}{\partial t} + \nabla.(\rho v \bar{v}) = -\frac{\partial p}{\partial y} + \rho f_y \\
&\frac{\partial(\rho w)}{\partial t} + \nabla.(\rho w \bar{v}) = -\frac{\partial p}{\partial z} + \rho f_z
\end{aligned} \qquad (1.2.14)$$

and

$$\frac{\partial}{\partial t}(\rho E) + \nabla.(\rho E \bar{v}) = -\nabla.(p \bar{v}) + \rho q + \rho(\bar{f} \cdot \bar{v})$$

Again, the equation of state, Eq. (1.1.8)

$$p = \rho R T$$

and the energy relation

$$e = e(T, v)$$

augment these equations.

1.2.4 Rotating Frame of Reference

In many applications it is necessary to describe the governing equations in a rotating frame of reference. If the system is rotating steadily with angular velocity $\bar{\omega}$ around an axis along which a coordinate z is aligned, and $\bar{w}$ is the velocity field relative to the rotating frame of reference, then

$$\bar{v} = \bar{w} + \bar{\omega} \times \bar{r} .$$

The rotating frame of reference does not effect the conservation of mass equation, but introduces additional terms into the momenta and energy equations. Full details of the complete equations can be found in standard texts[3,4].

1.2.5 Physics of Inviscid Flows

It is important to understand both in a mathematical and a physical sense the consequences of neglecting the viscous stresses inherent to any fluid. Reviewing the governing equations it is noted that the temperature only appears in the form of the equation of state and that the flow is governed by a system of non-linear partial differential equations of first order.

The equations describe flows with or without rotation and it is appropriate to introduce a relationship known as Crocco's equation which, for steady flows, can be written as

$$\bar{v} \times \bar{\omega} = \nabla H - T \nabla S$$

where $\bar{v}$ is the velocity vector, $\bar{\omega}$ the vorticity vector, H the total energy and S the entropy. Thus, neglecting viscosity and heat conduction, vorticity is present in the field of flow whenever the distribution of the total energy H or the entropy S is not uniform. This can happen, for example, when the fluid starts from a state of rest but of non-uniform temperature, or downstream of a curved shock wave. If vorticity is present and/or is created, then the convective terms in the Euler equations ensure that it will be convected around the flow field. It also implies that once vorticity has been created it is modelled in a mathematically consistent form. The ability of the equations to admit vorticity is important in such applications as flows involving jets, flows involving propeller slip streams and rotating systems like propellors and helicopter rotors.

The uniqueness of a solution derived from the Euler equations is ensured by imposing the additional condition that entropy may not decrease along a streamline (Second Law of Thermodynamics). This then precludes the existence of expansion shocks.

The Euler equations admit 'weak' solutions with contact or vortex sheet type discontinuities. This raises the interesting issue of how the equations model lift. In the Full Potential equation it is necessary to introduce a cut carrying a jump in potential from a trailing edge to downstream infinity in order to model the circulation in each section of a wing. By contrast, however, since vortex sheet discontinuities represent possible weak solutions, it can be argued that circulation and hence lift are phenomena that are inherently modelled by the Euler equations.

Much discussion has taken place on such issues and many of the issues have been resolved through numerical experimentation. It is now known that, with suitable artificial viscosity models, vortex sheets can be captured like shock waves and that in a time dependent solution approach (with artificial viscosity) the circulation and lift of airfoils and wings come out at the correct level without having to impose a Kutta condition. It is now thought that this situation arises because the artificial viscosity plays a similar role to real time dependent viscosity. So, although the Euler equations do not have a mechanism for the generation of vorticity, apart from at shock waves, the artificial viscosity inherent within the numerical solution of the equations plays a role similar to the physical viscosity in the sense that it generates vorticity which can cause, for example, the flow to separate.

It can also be argued that an inviscid shear flow negotiating a pressure gradient will separate when the static pressure equals the total pressure of the surface streamline. It therefore seems reasonable that we may expect solutions of the Euler equations with added artificial viscosity to exhibit separation and vortex sheets when i) sufficient vorticity has been generated through the artificial viscosity (in particular, at the surface of a configuration) and ii) there are sufficiently large variations in flow angle and/or pressure gradients. However, there is a general lack of knowledge on the behaviour of the mathematical solutions of the Euler equations near non smooth boundary surfaces.

The amount of vorticity generated by the artificial viscosity is dependent upon many aspects of the numerical procedure, including mesh characteristics and imposition of boundary conditions. In the case of flow around a sharp edge the positive and adverse pressure gradients are so large that the 'inviscid' separation always takes place. On smooth surfaces 'inviscid' separation may or may not take place, depending on the amount of vorticity generated by the artificial viscosity and local adverse pressure gradient.

Problems and issues raised here may be considered in the more general context of existence and uniqueness of steady solutions of the Navier-Stokes equations in the limit of vanishing viscosity. The conditions for existence and uniqueness of such solutions are not generally known. However, there is experimental evidence that for finite but high Reynolds numbers there may be conditions where a steady flow solution does not exist or where there may be multiple steady solutions.

1.2.6 References

1. DESHPANDE S. M. "On the Maxwellian Distribution, Symmetric Form, and Entropy Conservation for the Euler Equations", NASA TP-2583, 1986.

2. ANDERSON J. D. *Modern Compressible Flow*, McGraw-Hill, 1989.

3. BATCHELOR G. K. *An Introduction to Fluid Dynamics*, Cambridge University Press, 1974.

4. HIRSCH C. *Numerical Computation of Internal and External Flows*, Wiley, 1988.

1.3 JUMP RELATIONS, WEAK SOLUTIONS, RIEMANN PROBLEM

Discontinuous solutions of the system of Euler equations may exist and develop due to the hyperbolic nature of these equations. They are actually contained in the integral conservation laws for an arbitrary volume Ω bounded by a surface Σ moving with velocity $\bar{u}_\Sigma$:

$$\frac{d}{dt} \int_\Omega \mathbf{w} dx + \int_\Sigma (\bar{\mathbf{F}} - \mathbf{w}\, \bar{u}_\Sigma) \cdot \bar{n}_\Sigma \, d\sigma = 0 \qquad (1.3.1)$$

Assuming that a discontinuity surface S moving with velocity $\bar{u}_S$ is present inside Ω and divides it in two volumes Ω_1 and Ω_2, we apply the conservation laws separately to each volume and subtract the result from Eq. (1.3.1), that leads to:

$$\int_S [(\bar{\mathbf{F}} - \mathbf{w}\, \bar{u}_S) \cdot \bar{n}_S] \, d\sigma = 0 \qquad (1.3.2)$$

where $[\Phi]$ stands for $(\Phi_2 - \Phi_1)$, the jump of Φ through S.

This integral being zero independently of the choice of Ω, we get :

$$[(\bar{\mathbf{F}} - \mathbf{w}\, \bar{u}_S) \cdot \bar{n}_S] = 0 \qquad (1.3.3)$$

Replacing $\bar{\mathbf{F}}$ by its expression we find the "jump relations" :

$$[\rho (\bar{v} - \bar{u}_S) \cdot \bar{n}] = 0 \qquad (1.3.4a)$$

$$[\rho \bar{v} (\bar{v} - \bar{u}_S) \cdot \bar{n} + p \bar{n}] = 0 \qquad (1.3.4b)$$

$$[\rho E (\bar{v} - \bar{u}_S) \cdot \bar{n} + p \bar{v} \cdot \bar{n}] = 0 \qquad (1.3.4c)$$

or, after introducing v_r, the normal relative velocity of the fluid on S

$$v_r = (\bar{v} - \bar{u}_S) \cdot \bar{n}$$

with v_{r_1} and v_{r_2} not necessarily equal, we get the Rankine Hugoniot relations :

$$[\rho v_r] = 0 \qquad (1.3.5a)$$

$$[\rho v_r \bar{v} + p \bar{n}] = 0 \qquad (1.3.5b)$$

$$[\rho E v_r + p \bar{v} \cdot \bar{n}] = 0 \qquad (1.3.5c)$$

Further developments follow from considering the value Q of the mass flow through S :

$$Q = \rho v_r = \rho_1 v_{r_1} = \rho_2 v_{r_2} \qquad (1.3.6)$$

with the decomposition of $\bar{v}$ into :

$$v_n = \bar{v} \cdot \bar{n} \quad \text{and} \quad \bar{v}_t = \bar{v} - v_n \bar{n}$$

Then, Eq. (1.3.5b) gives :

$$Q [v_n] + [p] = 0 \qquad (1.3.7)$$

$$Q [\bar{v}_t] = 0 \qquad (1.3.8)$$

and Eq. (1.3.5c) gives :

$$Q [E] + [p v_n] = 0 \qquad (1.3.9)$$

or

$$Q [H] + [p] \bar{u}_S \cdot \bar{n} = 0 \qquad (1.3.10)$$

since $H = (E + p / \rho)$.

The discussion about the properties of a discontinuous solution of the Euler system is based on Eq. (1.3.7-9) and depends on the value of Q.

1.3.1 Contact discontinuity - Vortex sheet

When $Q = 0$, the mass flow through S is zero, S acts like an impervious surface with continuity of the pressure since from Eq. 1.3.7 we have :

$$[p] = 0$$

It is also clear from Eq. (1.3.9) that

$$[v_n] = 0$$

However $[\bar{v}_t]$ is arbitrary and we can distinguish two situations :

$$a) [\bar{v}_t] \neq 0$$

We have $v_{n_1} = v_{n_2}$ but $\bar{v}_{t_1} \neq \bar{v}_{t_2}$ and the discontinuity is a "vortex sheet".

$$b) [\bar{v}_t] = 0$$

Then $[\bar{v}] = 0$, velocity and pressure are continuous across S whereas ρ, E, s, H may have a jump through this "contact discontinuity".

1.3.2 Shock surfaces

We assume that $Q \neq 0$ then from Eq. (1.3.8) we find that

$$[\bar{v}_t] = 0$$

The case of $[v_n] = 0$ gives the trivial situation where there is not any discontinuity through S :

$$[\bar{v}_t] = 0, [\rho] = [p] = [E] = 0$$

It appears that we must have $[v_n] \neq 0$ so that ρ, p, E are discontinuous through S. From Eq. (1.3.7) and Eq. (1.3.10) we can calculate the normal velocity of the shock :

$$u_S = \bar{u}_S \cdot \bar{n} = \frac{[H]}{[\bar{v} \cdot \bar{n}]}$$

We note also that for a steady shock $[H] = 0$. Therefore, in a steady flow, H is constant on streamlines and when H is uniform at infinity the energy equation can be replaced by the Bernoulli equation :

$$H = H_0$$

From the second principle of thermodynamics one can write:

$$\frac{d}{dt} \int_\Omega \rho s \, dx + \int_\Sigma \rho s (\bar{v} - \bar{u}_\Sigma) \cdot \bar{n}_\Sigma \, d\sigma \geq 0 \qquad (1.3.11)$$

and as for conservation laws we derive the following condition for the entropy jump through a shock S :

$$[\rho s (\bar{v} - \bar{u}_S) \cdot \bar{n}_S] \geq 0 \quad \text{or}$$

$$Q [s] \geq 0 \qquad (1.3.12)$$

It can be proved that $[s] = 0$ should imply the continuity of all variables thus

$$Q\,[\,s\,] > 0$$

The entropy must increase through a shock along the flow direction and from the Rankine-Hugoniot relations, we deduce:

$$Q\,[\,p\,] > 0\,, \quad Q\,[\,\rho\,] > 0.$$

Only a compression can occur in a physical discontinuity and unphysical "expansion shocks" are ruled out by Eq. (1.3.12)

1.3.3 Weak solutions and uniqueness

The system of Euler equations in conservative differential form Eq. (1.2.14) has been derived from the integral conservation laws by assuming continuity and derivability of the physical variables. One can consider that discontinuous solutions of Eq. (1.2.14) exist only in the sense of distributions.

To formalize this notion it is necessary to introduce a weak formulation of the Euler system of equations. Each equation in differential form is first multiplied by a "test function" ϕ infinitely differentiable, then space and time integration by parts allows to free physical unknowns from any partial derivation. We get :

$$\int_{t_0}^{t_1} \int_{\Omega} \mathbf{w}_i \frac{\partial \phi}{\partial t} + \vec{\mathbf{F}}_i \cdot \vec{\nabla}\phi \, dx \, dt =$$

$$- \int_{t_0}^{t} \int_{\Omega} g_i \phi \, dx \, dt \;, \text{ for all test functions } \phi \qquad (1.3.13)$$

where g_i takes into account the external forces.

It can be checked that if $S\,(x\,,\,t\,) = 0$ is the equation of a surface on which $\mathbf{w}$ and $\vec{\mathbf{F}}\,(\mathbf{w}\,)$ are discontinuous we find from Eq. (1.3.13) :

$$[\,\mathbf{w}_i\,] \frac{\partial S}{\partial t} + [\,\vec{\mathbf{F}}_i\,] \vec{\nabla} S = 0$$

and since $\vec{n}_S = - \dfrac{\vec{\nabla} S}{|\vec{\nabla} S|}$ and $\vec{u}_S = - \dfrac{\partial S}{\partial t} \dfrac{\vec{\nabla} S}{|\vec{\nabla} S|^2}$

We obtain Eq (1.3.3) :

$$[\,(\,\vec{\mathbf{F}}_i - \mathbf{w}_i\,\vec{u}_S\,)\,.\,\vec{n}_S\,] = 0 \qquad i = 1, \ldots, m$$

The set of weak solutions obeying Eq. (1.3.13) comprises discontinuous solutions with both physical and non physical discontinuities. As said above, the non physical discontinuities are the so-called "expansion shocks" satisfying the Rankine Hugoniot relations, they correspond locally to a flow with a valid compression shock in which the direction of velocity should have been reversed. It is necessary, in order to discard these non physical weak solutions and thus to avoid non uniqueness problems, to take into account the entropy inequality.

Theoretical studies have been done on hyperbolic systems of conservation laws in order to complement them by an entropy inequality in divergence form with the definition of an entropy function and of the corresponding entropy fluxes ensuring that an entropy condition holds (Godunov[1], Lax[2], Harten[3]). Interesting consequences of these studies are the introduction of "entropy variables" mainly used in the finite element methods (see below Section 2.1) and recent studies on discrete entropy inequalities and entropic schemes (Tadmor[4], Lerat[5]).

In order to discuss the uniqueness of an Euler flow solution, it is necessary not only to look at the discontinuities and on the entropy inequality for the flowfield but also to give precise information on the choice of the boundary conditions in the farfield or at the entrance and exit boundaries and on a wall boundary. In fact, little can be said for uniqueness of solutions in the large, except that the choice and the role of boundary conditions is very important and it suffices to look at the problem of a subsonic flow past an ellipse at a given angle of attack (T.H. Pulliam[6]) or at a vortical flow with closed streamlines in two dimensions to see the difficulty of this question.

1.3.4 The Riemann problem

The Riemann problem consists of the initial value problem in one space dimension with initial data (at $t = 0$) given by two constant states $(\,\mathbf{w}_L\,,\,\mathbf{w}_R\,)$ separated by an arbitrary discontinuity located at $x = 0$. This problem is of special interest since it represents (with the inviscid assumption) the unsteady flow in a "shock tube" after the bursting of a diaphragm and moreover its solution can be found as a system of simple waves with only a single nonlinear equation to be solved. Another very important reason for introducing the Riemann problem is its current use in the design of numerical schemes for solving Euler equations since the pioneering scheme of Godunov[7].

To study the Riemann problem it is convenient to use not only the conservative variables but also (ρ, v, p) variables.

The initial data are : $\mathbf{u}_L = (\,\rho_L\,,\,v_L\,,\,p_L\,)$ and $\mathbf{u}_R = (\,\rho_R\,,\,v_R\,,\,p_R\,)$ representing respectively left and right constant states.

At $t > 0$, the solution depends only on x/t (it is auto-similar) and is made (for a typical shock tube problem with $p_L > p_R$) of a rarefaction wave (1-wave), a contact discontinuity (2-wave) and a shock (3-wave), with :

$$\lambda_1 = v - a\,, \quad \lambda_2 = v\,, \quad \lambda_3 = v + a\,,$$

the eigenvalues of the Jacobian $d\vec{\mathbf{F}}/d\mathbf{w}$ each associated with a characteristic curve and a Riemann invariant.

Two constant states appear between the expansion fan and the shock : $(\,\rho_L^*\,,\,v^*\,,\,p^*\,)$ and $(\,\rho_R^*\,,\,v^*\,,\,p^*\,)$.

Note the continuity of the velocity and pressure across the contact discontinuity. The behaviour of the solution is described in Fig. 1.3.1.

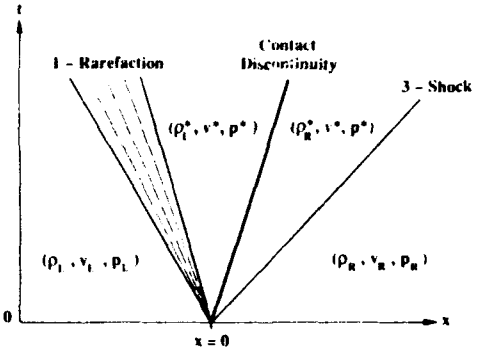


Figure 1.3.1 The Riemann problem

The details of solving the Riemann problem for the general case can be found in many text books (see for example Courant and Friedrichs[8]).

We give below only some indications on the way towards the solution.

From $(\gamma - 1)/2\ v + a = cst$ on a 3-characteristics and from $s = cst$ on a 2-characteristics both crossing the rarefaction fan (1-wave), a relation can be derived giving v^* as a (non-linear) function of p^*.

From the Rankine-Hugoniot relations through the shock another equation can be derived giving v^* in terms of p^*. Elimination of v^* leads to a non-linear equation to be solved for p^* by Newton iteration. Then v^* is calculated and also ρ_L^* and ρ_R^*.

Finally, in the expansion fan made of straight 1-characteristics of slope : $\xi = x/t = v - a$, v and a are both constant for a given ξ.

By use of $(\gamma - 1)/2\ v + a = cst$ in the expansion fan, v is expressed as a linear function of ξ and so is a. The pressure is obtained through the isentropic relation in terms of $M^2 = v^2 / a^2$.

1.3.5. References

1. GODUNOV, S.K. "An interesting Class of Quasilinear Systems", Dokl. Acad. Nauk, SSSR, 139 (1961), pp 521-523.

2. LAX, P.D. "*Hyperbolic Systems of Conservation Laws and the Mathematical Theory of Shock Waves*", CBMS Regional Conference Series in Applied Mathematics 11, SIAM, Philadelphia, 1972.

3. HARTEN, A. "On the Symmetric Form of Systems of Conservation Laws with Entropy", ICASE Report 81-34, ICASE Hampton, VA, 1981.

4. TADMOR, E. "The Numerical Viscosity of Entropy Stable Schemes for Systems of Conservation Laws, I", Math. Computat., 49, pp 91-103, 1987.

5. KHALFALLAH, K. and LERAT, A. "Correction d'entropie pour les schémas numériques approchant un système hyperbolique", C.R. Acad. Sc. Paris, 308 II, pp 815-820, 1989.

6. PULLIAM, T.H. "Computational Challenge: Euler Solution for Ellipses", AIAA Journal, Vol 28, NO 10, Oct. 1990.

7. GODUNOV, S.K. "A Finite Difference Method for the Numerical Computation of Discontinuous Solutions of the Equations of Fluid Dynamics", Math. Sbornik, Vol 47, pp 357-398, 1959.

8. COURANT, R. and FRIEDRICHS, K.O. "*Supersonic Flow and Shock Waves*", Wiley-Interscience, New York, 1948.

1.4 BOUNDARY CONDITIONS

The time-dependent Euler equations are a hyperbolic system of equations. Numerical computations must be done on a finite mesh and, thus, waves which are incoming and outgoing with respect to the computational domain cross the boundary of the mesh. Application of the interior point algorithm at the boundary requires information from outside the domain, which is generally not completely available. A characteristic decomposition at the boundary indicates that the equations representing outgoing waves can be differenced using information from the computational domain. The equations representing incoming waves cannot be stably differenced using only information available in the interior; hence, those equations need to be replaced by boundary conditions. Thus, the number of boundary conditions to be specified at the boundary comes from a straightforward characteristic analysis at the boundary.

The type of boundary condition to be specified is problem dependent and as Moretti[1] noted: "A physically consistent model of the outside world must be provided." For instance, the upstream boundary conditions to be specified for a subcritical two-dimensional converging-diverging nozzle might consist of the recognition that the fluid comes from a uniform reservoir condition, corresponding to specified total pressure and entropy conditions, as well as a specification of the direction of the velocity. Likewise, since the mass flow through a one-dimensional nozzle is known to be set by the back pressure, a valid downstream boundary condition is the specification of the pressure. The variables at the boundary can be constructed from these boundary conditions, supplemented by characteristic equations corresponding to the outgoing waves applied at the boundary.

1.4.1 Characteristic Equations

Linearized Equations

The conservation law form of the time-dependent Euler equations are written here as

$$\frac{\partial \mathbf{w}}{\partial t} + \frac{\partial \mathbf{F}}{\partial x} + \frac{\partial \mathbf{G}}{\partial y} + \frac{\partial \mathbf{H}}{\partial z} = 0 \tag{1.4.1}$$

The linearized form of the equations can be written as

$$\frac{\partial \mathbf{w}}{\partial t} + \mathbf{A}\frac{\partial \mathbf{w}}{\partial x} + \mathbf{B}\frac{\partial \mathbf{w}}{\partial y} + \mathbf{C}\frac{\partial \mathbf{w}}{\partial z} = 0 \tag{1.4.2}$$

where $\mathbf{A}, \mathbf{B}, \mathbf{C}$ are Jacobian matrices (i.e., $\mathbf{A} = \partial\mathbf{F}/\partial\mathbf{w}$). The linearized equations can be cast in terms of a set of *primitive* variables using chain-rule differentiation as

$$\mathbf{M}\frac{\partial \mathbf{q}}{\partial t} + \mathbf{AM}\frac{\partial \mathbf{q}}{\partial x} + \mathbf{BM}\frac{\partial \mathbf{q}}{\partial y} + \mathbf{CM}\frac{\partial \mathbf{q}}{\partial z} = 0 \tag{1.4.3}$$

$$\frac{\partial \mathbf{q}}{\partial t} + \mathbf{a}\frac{\partial \mathbf{q}}{\partial x} + \mathbf{b}\frac{\partial \mathbf{q}}{\partial y} + \mathbf{c}\frac{\partial \mathbf{q}}{\partial z} = 0 \tag{1.4.4}$$

where $\mathbf{M} = \partial\mathbf{w}/\partial\mathbf{q}$, $\mathbf{M}^{-1} = \partial\mathbf{q}/\partial\mathbf{w}$, and $\mathbf{a}=\mathbf{M}^{-1}\mathbf{AM}$,.... The choice of primitive variables is not unique and is generally selected to make the Jacobian matrices **a**,**b**,**c** simpler than their counterparts using the conserved variables.[2,3] A common choice is the set

$$\mathbf{q} = \begin{bmatrix} \rho \\ u \\ v \\ w \\ p \end{bmatrix} \tag{1.4.5}$$

for which

$$\mathbf{M} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ u & \rho & 0 & 0 & 0 \\ v & 0 & \rho & 0 & 0 \\ w & 0 & 0 & \rho & 0 \\ \varphi^2/2 & \rho u & \rho v & \rho w & 1/\kappa \end{bmatrix} \tag{1.4.6}$$

$$\mathbf{M}^{-1} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ -u/\rho & 1/\rho & 0 & 0 & 0 \\ -v/\rho & 0 & 1/\rho & 0 & 0 \\ -w/\rho & 0 & 0 & 1/\rho & 0 \\ \kappa\varphi^2/2 & -\kappa u & -\kappa v & -\kappa w & \kappa \end{bmatrix} \tag{1.4.7}$$

where $\kappa = \gamma - 1$, $\varphi^2 = u^2 + v^2 + w^2$. The Jacobian matrix **a**, for this choice, is

$$\mathbf{a} = \begin{bmatrix} u & \rho & 0 & 0 & 0 \\ 0 & u & 0 & 0 & 1/\rho \\ 0 & 0 & u & 0 & 0 \\ 0 & 0 & 0 & u & 0 \\ 0 & \rho a^2 & 0 & 0 & u \end{bmatrix} \tag{1.4.8}$$

from which the eigenvalues of **a** can be easily computed as

$$u, u, u, u + a, u - a \tag{1.4.9}$$

All of the Jacobian matrices have real eigenvalues and a set of linearly independent eigenvectors and each, individually, can be diagonalized, although not simultaneously since the Jacobian matrices do not have the same eigenvectors. The characteristic equations result from diagonalizing the Jacobian matrices as

$$\frac{\partial \mathbf{q}}{\partial t} + \mathbf{T}_x\mathbf{\Lambda}_x\mathbf{T}_x^{-1}\frac{\partial \mathbf{q}}{\partial x} + \mathbf{T}_y\mathbf{\Lambda}_y\mathbf{T}_y^{-1}\frac{\partial \mathbf{q}}{\partial y} + \mathbf{T}_z\mathbf{\Lambda}_z\mathbf{T}_z^{-1}\frac{\partial \mathbf{q}}{\partial z} = 0 \tag{1.4.10}$$

where $\mathbf{\Lambda}_x, \mathbf{\Lambda}_y, \mathbf{\Lambda}_z$ are diagonal matrices.

Considering a plane boundary as coincident with a surface of constant x, the derivatives in the two directions tangent to the boundary, y, z, can be determined from information on the boundary. In general, the computation of the derivative normal to the boundary requires information about the state vector at locations outside the computational domain. Defining the terms corresponding to derivatives in the plane of the boundary as a source term **S**, the equations may be written as[3,4]

$$\mathbf{T}^{-1}\frac{\partial \mathbf{q}}{\partial t} + \mathbf{\Lambda}\mathbf{T}^{-1}\frac{\partial \mathbf{q}}{\partial x} + \mathbf{T}^{-1}\mathbf{S} = 0 \tag{1.4.11}$$

where the x-subscript notation has been dropped. This can also be written in terms of each component of the equation as

$$\mathbf{l}_i\frac{\partial \mathbf{q}}{\partial t} + \lambda_i\mathbf{l}_i\frac{\partial \mathbf{q}}{\partial x} + \mathbf{l}_i\mathbf{S} = 0 \tag{1.4.12}$$

where $\mathbf{l}_i$ is the left eigenvector of the Jacobian matrix **a**, corresponding to the ith eigenvalue (and also forms the ith row of $\mathbf{T}^{-1}$).

Characteristic Variables

If a characteristic variable V_i can be defined to satisfy the so-called compatibility equations below

$$dV_i = \mathbf{l}_i d\mathbf{q} + \mathbf{l}_i\mathbf{S}dt \tag{1.4.13}$$

then Eq. (1.4.12) reduces to a set of wave equations

$$\frac{\partial V_i}{\partial t} + \lambda_i \frac{\partial V_i}{\partial x} = 0 \tag{1.4.14}$$

for which the characteristic variable is constant along characteristic curves defined in the $x-t$ plane as $dx/dt = \lambda_i$.[2,5] The characteristic directions are sketched in Fig. 1.4.1 for subsonic and supersonic flow. Positive u is indicated; outflow conditions correspond to the exterior boundary defined by $x > 0$. The characteristics are traced back from the new time level t^{n+1} for the three characteristic directions, C^0, C^+, C^-, corresponding to the repeated eigenvalue λ_1 and λ_4, λ_5.

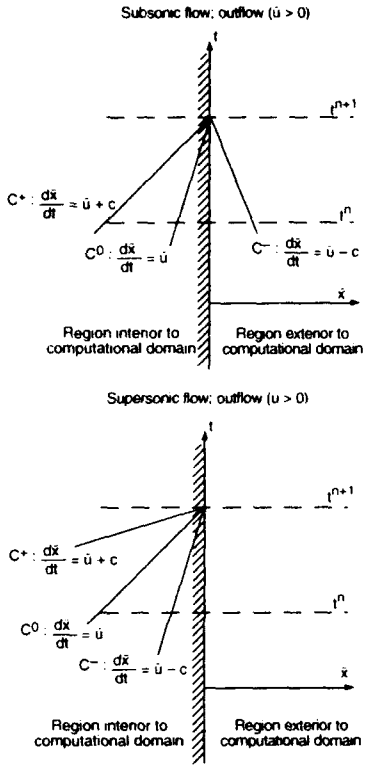


Figure 1.4.1 Sketch of characteristic directions at the boundary. Outflow (inflow) corresponds to the exterior domain described by $x > 0$ ($x < 0$).

The construction of V_i is generally not possible for the Euler equations without assuming the diagonalizing matrices are constant.[2,3] Assuming the exterior domain is described by $x > 0$, the characteristic form, either Eq. (1.4.12) or (1.4.14), indicates that outgoing waves are described by equations with $\lambda_i \geq 0$ and depend on information at and within the boundary. Incoming waves, representing information reaching the boundary from the exterior, are described by equations with $\lambda_i < 0$. These wave equations cannot be differenced stably using just interior and boundary information since the numerical domain of dependence would not include the physical domain of dependence; hence, these equations need to be replaced with boundary conditions.

Diagonal Equations

The diagonalizing matrices in the above case are given as

$$\mathbf{T}^{-1} = \begin{bmatrix} 1 & 0 & 0 & 0 & -1/a^2 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1/\rho a \\ 0 & -1 & 0 & 0 & 1/\rho a \end{bmatrix} \tag{1.4.15}$$

$$\mathbf{T} = \begin{bmatrix} 1 & 0 & 0 & \rho/2a & \rho/2a \\ 0 & 0 & 0 & 1/2 & -1/2 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & \rho a/2 & \rho a/2 \end{bmatrix} \tag{1.4.16}$$

Using the primitive variable diagonalization results, the conserved variable Jacobians can be diagonalized easily as

$$\mathbf{A} = \mathbf{M}\mathbf{T}\boldsymbol{\Lambda}\mathbf{T}^{-1}\mathbf{M}^{-1} \tag{1.4.17}$$

Assuming no variations in the plane of the boundary and the diagonalizing matrices to be constant, the equations can be reduced to a set of diagonal equations

$$\frac{\partial \bar{\mathbf{w}}}{\partial t} + \boldsymbol{\Lambda}\frac{\partial \bar{\mathbf{w}}}{\partial x} = 0 \tag{1.4.18}$$

where the linearized characteristic variable is defined as

$$\bar{\mathbf{w}} = \mathbf{T}_o^{-1}\mathbf{q} = \begin{bmatrix} \rho - p/a_0^2 \\ v \\ w \\ u + p/\rho_0 a_0 \\ -u + p/\rho_0 a_0 \end{bmatrix} \tag{1.4.19}$$

and the subscript notation denotes evaluation at a nearby reference value. Note that the linearized characteristic variables can also be cast in terms of the conserved variable vector[6] as $\bar{\mathbf{w}} = \mathbf{T}_o^{-1}\mathbf{M}_0^{-1}\mathbf{w}$.

Homentropic Equations

Assuming locally homentropic flow (i.e., that the entropy is uniform everywhere) and no spatial variations in the plane of the boundary, the equations can be reduced to

$$\frac{d}{dt}(v) = 0 \quad \text{along} \quad \frac{dx}{dt} = u \tag{1.4.20}$$

$$\frac{d}{dt}(w) = 0 \quad \text{along} \quad \frac{dx}{dt} = u \tag{1.4.21}$$

$$\frac{d}{dt}(R^{\pm}) = 0 \quad \text{along} \quad \frac{dx}{dt} = u \pm a \tag{1.4.22}$$

where the Riemann variables are defined as

$$R^{\pm} = u \pm [2a/(\gamma - 1)] \tag{1.4.23}$$

and x is the local normal pointing out of the domain. The equations are in a form very similar to one-dimensional unsteady flow, except that the tangential velocities, in addition to the entropy, are convected along the particle path.

1.4.2 Numerical Procedures

The equations which replace the characteristic equations for the incoming waves are generally referred to as the physical boundary conditions. The procedures to determine the remaining variables at the boundary, which should be as compatible as possible with the outgoing characteristic equations, are sometimes referred to as numerical boundary conditions, but should be more properly termed numerical treatments at the boundary.[1]

The numerical procedures at the boundary are different than those used at the interior scheme. Thus, two factors of the coupled system need to be taken into account: the accuracy and the stability. Gustafsson[7] has pointed out that the accuracy of the numerical procedure for a linear equation can be one order lower than the order of the interior scheme without adversely influencing the global accuracy of the solution. The stability of the boundary procedure can be analyzed in many cases using the analysis of Gustafsson, Kreiss, and Sundstrom.[8] Generally, the closer the numerical scheme is coupled to the characteristic equations, the more well-behaved the numerical procedure.

The equations and procedures for a stationary boundary analysis are indicated here, where a local orthogonal coordinate system is assumed at the boundary. This is consistent with the methods in common use, although the choice of coordinate system is not unique and need not be taken normal to the boundary, as pointed out by Roe.[9] The extension to a moving boundary can be accomplished in a straightforward manner. The eigenvalues of the Jacobian matrices are changed by the addition of a term which is the speed of the grid normal to the boundary. However, the eigenvectors and, hence, the basic character of the equations are unchanged from those for the stationary generalized coordinate system.[4,10]

Characteristic Methods

The characteristic equations dictate that the equations corresponding to the incoming waves be replaced with boundary conditions as

$$B_j = 0 \qquad j = 1, N_I \tag{1.4.24}$$

where N_I is the number of incoming waves and N is the total number of equations. Thus, numerical procedures are required at the boundary, in general, to solve the N_I physical boundary condition equations and the $N - N_I$ outgoing characteristic equations. Chakravarthy[10] developed a unified approach, in which the incoming-wave equations are replaced with time-dependent boundary condition equations and solved numerically in a way consistent with the interior point scheme.

For example, at a subsonic outflow boundary ($N_I = 1$), the equation associated with the $\lambda_1 = u - a$ eigenvalue can be replaced with a linearized form of a time-invariant physical boundary condition

$$\frac{\partial B_1}{\partial \mathbf{q}} \frac{\partial \mathbf{q}}{\partial t} = 0 \tag{1.4.25}$$

An example and often-used boundary condition is the specification of pressure as:

$$B_1 = p - p_\infty = 0 \tag{1.4.26}$$

in which case $\partial B_1 / \partial \mathbf{q} = (0, 0, 0, 0, 1)$. Thus, the characteristic equations become

$$\begin{bmatrix} \mathbf{l}_1 \\ \mathbf{l}_2 \\ \mathbf{l}_3 \\ \mathbf{l}_4 \\ \partial B_1 / \partial \mathbf{q} \end{bmatrix} \frac{\partial \mathbf{q}}{\partial t} + \begin{bmatrix} \lambda_1 \mathbf{l}_1 \\ \lambda_2 \mathbf{l}_2 \\ \lambda_3 \mathbf{l}_3 \\ \lambda_4 \mathbf{l}_4 \\ 0 \end{bmatrix} \frac{\partial \mathbf{q}}{\partial x} + \begin{bmatrix} \mathbf{l}_1 \\ \mathbf{l}_2 \\ \mathbf{l}_3 \\ \mathbf{l}_4 \\ 0 \end{bmatrix} \mathbf{S} = 0 \tag{1.4.27}$$

which can serve to define the two matrices $\mathbf{P}_1, \mathbf{P}_2$ as

$$\mathbf{P}_1 \frac{\partial \mathbf{q}}{\partial t} + \Lambda \mathbf{P}_2 \frac{\partial \mathbf{q}}{\partial x} + \mathbf{P}_2 \mathbf{S} = 0 \tag{1.4.28}$$

$$\frac{\partial \mathbf{q}}{\partial t} + \mathbf{P}_1^{-1} \Lambda \mathbf{P}_2 \frac{\partial \mathbf{q}}{\partial x} + \mathbf{P}_1^{-1} \mathbf{P}_2 \mathbf{S} = 0 \tag{1.4.29}$$

The above equation can then be differenced at the boundary using one-sided derivatives. It can also be expressed as an equation in the conserved variables as

$$\frac{\partial \mathbf{w}}{\partial t} + \mathbf{M} \mathbf{P}_1^{-1} \mathbf{P}_2 \mathbf{M}^{-1} \frac{\partial \mathbf{F}}{\partial x} + \mathbf{M} \mathbf{P}_1^{-1} \mathbf{P}_2 \mathbf{S} = 0 \tag{1.4.30}$$

Either equation can be advanced in time explicitly or implicitly at the boundary; generally, the choice is made on the basis of compatibility with the interior point scheme.

Nonreflecting Methods

A closely related procedure is the so-called nonreflecting, or radiation, boundary conditions of Hedstrom[11] and Thompson.[3] In this approach, the amplitudes of the incoming waves are taken as constant, in time, at the boundary. This corresponds to specifying that the incoming characteristic variable is specified at the boundary, or

$$\frac{\partial V_i}{\partial t} = 0 \qquad i = 1, N_I \tag{1.4.31}$$

In terms of the subsonic outflow condition example above, the characteristic equations become

$$\begin{bmatrix} \mathbf{l}_1 \\ \mathbf{l}_2 \\ \mathbf{l}_3 \\ \mathbf{l}_4 \\ \mathbf{l}_5 \end{bmatrix} \frac{\partial \mathbf{q}}{\partial t} + \begin{bmatrix} \lambda_1 \mathbf{l}_1 \\ \lambda_2 \mathbf{l}_2 \\ \lambda_3 \mathbf{l}_3 \\ \lambda_4 \mathbf{l}_4 \\ 0 \end{bmatrix} \frac{\partial \mathbf{q}}{\partial x} + \begin{bmatrix} \mathbf{l}_1 \\ \mathbf{l}_2 \\ \mathbf{l}_3 \\ \mathbf{l}_4 \\ \mathbf{l}_5 \end{bmatrix} \mathbf{S} = 0 \tag{1.4.32}$$

This can be written as below, where $\mathbf{A}^+$ is the diagonal matrix composed of the nonnegative eigenvalues of $\mathbf{a}$:

$$\mathbf{T}^{-1} \frac{\partial \mathbf{q}}{\partial t} + \mathbf{A}^+ \mathbf{T}^{-1} \frac{\partial \mathbf{q}}{\partial x} + \mathbf{T}^{-1} \mathbf{S} = 0 \tag{1.4.33}$$

The nonreflecting characteristic equations can also be written in terms of conserved variables as

$$\frac{\partial \mathbf{w}}{\partial t} + \mathbf{A}^+ \frac{\partial \mathbf{w}}{\partial x} + \mathbf{M} \mathbf{S} = 0 \tag{1.4.34}$$

where the spatial difference can be evaluated using one-sided differencing at the boundary. The above equation appears as a nonconservative scheme evaluated locally at the boundary, where the strength of the incoming wave is defined to be zero. With the equations in this form, it is apparent that any upwind scheme can be used to define a nonreflecting operator at the boundary by defining the strengths of the incoming waves to be zero.

Rudy and Strikwerda[12] recognized that this procedure, designed to minimize reflections from the boundary, is usually not consistent with physical boundary conditions defined on the basis of steady-state ideas. For example, the specification of constant pressure in the subsonic outflow case above ensures that a wave is reflected back into the interior. Rudy and Strikwerda added a parameter-dependent source term so that the solution to the discrete equations went to a specified value at steady state. More recent developments in nonreflecting boundary conditions are given by Giles;[13] quasi-three-dimensional applications to internal cascade flow are given by Saxer and Giles.[14] Atkins and Casper[15] developed a boundary condition procedure by connecting the boundary values to the uniform far-field conditions and the interior conditions through several simple wave fields and demonstrated an improved calculation for one-dimensional wave propagations using higher order discretizations.

Analytic Methods

An alternative procedure has been developed by Verhoff et al.[16] and Hirsch and Verhoff.[17] It is a consistent method for coupling linearized analytic solutions with nonlinear numerical solutions through the computational boundary condition. While limited to steady flow, the procedure is derived from an asymptotic expansion to the Euler equations and, thus, is more general than a far-field potential correction method, since the method can treat strong shocks and rotational flows. The analysis is based on a linearization about a true uniform steady state valid at far distances with the equations cast in terms of Riemann–like variables using a streamline coordinate system. The linearized characteristic equations representing incoming waves are solved in the exterior domain using a Fourier transform technique; the solution involves integration along computational boundaries, which are taken as a parabola at inflow and a straight line at outflow. Two-dimensional procedures and results for internal and external flows have been obtained;[16,17] an example is given subsequently in this section.

Extrapolation Methods

Simpler procedures are also used frequently which are based on honoring the domain of dependence of the characteristic equations. For example, the characteristic variables, evaluated at local conditions, are often extrapolated to the boundary instead of solving the characteristic equations.[4]

Homentropic Methods

Assuming a locally orthogonal coordinate system where $\bar{x}$ is the local normal pointing out of the domain (sketched in Fig. 1.4.2), then the homentropic equations, Eqs. (1.4.20)–(1.4.22), can be used to update the equations along the boundary at the new time level. For subsonic flow, for instance, R^- can

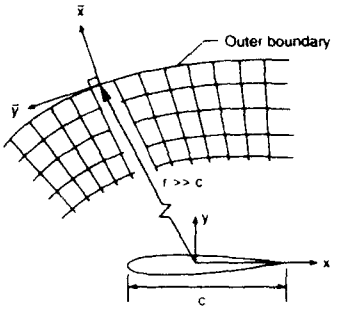


Figure 1.4.2 Sketch of local coordinate system at the boundary used for characteristic method in homentropic flow.

be evaluated from the far field, corresponding to conditions outside the boundary, and R^+ can be evaluated locally from the interior of the domain. Then, the normal velocity and speed of sound can be evaluated as

$$u = (R^+ + R^-)/2 \tag{1.4.35}$$

$$a = (R^+ - R^-)(\gamma - 1)/4 \tag{1.4.36}$$

Depending on the sign of the normal velocity, the entropy and tangential velocities are extrapolated from the exterior or interior of the domain. Thus, the three velocity components, entropy, and speed of sound can be constructed at the new time level. Note that the state vector can be determined without an explicit construction of the tangential boundary direction cosines. Denoting the velocity components corresponding to the region from which the entropy and tangential velocities are extrapolated as u, v, w (these are free-stream velocities for inflow conditions at the boundary), the velocity vector at the new time level is

$$u^{n+1} = u + \mathbf{n}_x(u - u) \tag{1.4.37}$$

$$v^{n+1} = v + \mathbf{n}_y(u - v) \tag{1.4.38}$$

$$w^{n+1} = w + \mathbf{n}_z(u - w) \tag{1.4.39}$$

Note that the procedure does not ensure the conservation of total enthalpy, and in some schemes the conservation of total enthalpy is an important feature of the interior point scheme. In those cases, some modification of the procedure is required,[18] such as the redefinition of the speed of sound from Eq. (1.4.36) to ensure constant total enthalpy.

1.4.3 Inflow/Outflow Boundaries

Supersonic Flow

For supersonic flow normal to the boundary, all of the characteristic directions are of the same sign. At inflow, all quantities should be specified. At outflow, the characteristic equations can be differenced in a one-sided manner using information in the computational domain. It is quite common, however, to just honor the domain of dependence constraint and extrapolate the state variable to the boundary from the interior.

Subsonic Flow

For subsonic flow normal to the boundary, four boundary conditions can be set at inflow and one at outflow. However, it is difficult to specify the boundary conditions accurately since the influence of the computed airfoil (body) is felt at large distances upstream and downstream; the assumption of uniform flow at the boundary necessitates the construction of a grid which extends quite far from the airfoil. By including the first-order effect of the circulation imposed by the airfoil to the state variable vector at locations exterior to the boundary, the boundary need not extend as far; thus, the computations can be restricted to a smaller domain with fewer grid points and/or less stretching. A far-field boundary correction is used in most transonic potential flow methods, derived from the linearized small-disturbance equation, although the early work of Murman and Cole[19] used an expansion of the nonlinear small-disturbance potential equation. For two-dimensional flow, the nondimensional velocity components in the far field can be given as

$$u = \cos\alpha + F\sin\theta \tag{1.4.40}$$

$$v = \sin\alpha - F\cos\theta \tag{1.4.41}$$

where

$$F = \frac{(c_l c/4\pi)(1/r)}{\left[1 - M_\infty^2 \sin^2(\theta - \alpha)\right]} \tag{1.4.42}$$

where r, θ are the radius and polar angle, respectively; the coordinate system is located at the center of lift (generally the quarter-chord for an airfoil) and the angle θ is defined positive clockwise from the chord line extended downstream of the trailing edge. With constant total enthalpy and constant entropy specified in the far field, the state vector can be constructed at regions outside the boundary of use in the chosen boundary condition procedure. An example of including this effect on the lift coefficient of an airfoil is shown in Fig. 1.4.3. Assuming a locally orthogonal coordinate system, where $\tilde{r}$ is the local normal pointing out of the domain (sketched in Fig. 1.4.2), then the homentropic equations, Eqs. (1.4.20)–(1.4.22), can be used to update the equations along the boundary at the new time level. Subcritical and supercritical cases are shown. Using free-stream conditions to evaluate far-field boundary contributions, the lift coefficient shows an inverse radial dependence on the boundary extent, which is the same functional dependence as the leading-order term in the far-field expansion. Updating the boundary conditions with the far-field contribution corrected as above, the sensitivity of the solution is dramatically reduced. The supercritical case shows a stronger dependence on the outer boundary extent, as expected, due to the increased lateral extent of the disturbances at the higher Mach number. The correction term, which scales on the lift, is effective in both cases. For subcritical cases, circulation and lift are both constant on lines which encircle the airfoil; for supercritical flow, the circulation varies in the near field since vorticity can be generated at shocks. The three-dimensional extension for the perturbations due to circulation for thin lifting wings is given by Klunker.[20]

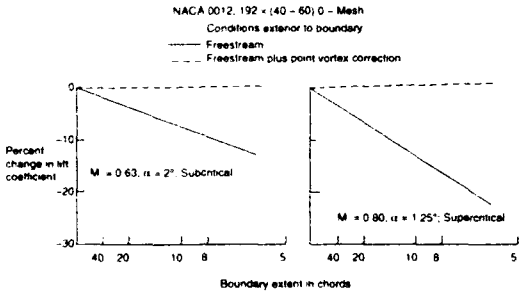


Figure 1.4.3 Effect of far-field boundary location on lift coefficient for NACA 0012 airfoil at subcritical and supercritical conditions.

A sample calculation using the analytic method of Verhoff et al.[16] is shown in Figs. 1.4.4–1.4.5 for the computation of the flow over a NACA 0012 airfoil at subcritical conditions. Computations were done using a baseline grid that extends 40 chords from the airfoil and a subset of the grid that extends only a small distance from the airfoil. The inner portion of the baseline grid is shown. The pressure contours from computations on the inner grid using free-stream (zero-order) and corrected (first-order) conditions exterior to the boundary are compared to calculations made with the complete baseline grid. The pressure contours of the inner grid calculation agree much closer when the first-order corrections are applied with those corresponding to the complete baseline grid.

1.4.4 Surface Boundaries

At a surface, the boundary condition is usually taken as tangency, so that the inviscid velocity component normal to the boundary is that of the boundary, which is normally zero. This is consistent with the characteristic analysis since only one wave is incoming at the boundary. One wave is outgoing and the rest travel along the boundary. The flux at the boundary simply reverts to the specification of the pressure

$$\mathbf{F} = \begin{bmatrix} 0 \\ p\mathbf{n}_x \\ p\mathbf{n}_y \\ p\mathbf{n}_z \\ 0 \end{bmatrix} \tag{1.4.43}$$

The pressure at the boundary can be determined from the outgoing characteristic relation. Alternately, the pressure can be extrapolated to the surface.

Normal Momentum Equation

Rizzi[21] replaced the outgoing characteristic relation with the normal momentum equation at the surface to determine the pressure. This is one of the most accurate and commonly used methods for determining pressure. For the analysis, a local coordinate system

$$(\xi, \eta, \zeta) \tag{1.4.44}$$

is used, where the boundary corresponds to a surface of constant η, for example. The normal momentum equation can be written in terms of the variations in the surface and the normal pressure gradient as

$$\frac{\partial p}{\partial \eta}(\eta_x^2 + \eta_y^2 + \eta_z^2) = -\frac{\partial p}{\partial \zeta}(\eta_x\zeta_x + \eta_y\zeta_y + \eta_z\zeta_z) - \frac{\partial p}{\partial \xi}(\eta_x\xi_x + \eta_y\xi_y + \eta_z\xi_z)$$

$$-\rho U\left(\eta_x \frac{\partial u}{\partial \xi} + \eta_y \frac{\partial v}{\partial \xi} + \eta_z \frac{\partial w}{\partial \xi}\right)$$
$$-\rho W\left(\zeta_x \frac{\partial u}{\partial \zeta} + \zeta_y \frac{\partial v}{\partial \zeta} + \zeta_z \frac{\partial w}{\partial \zeta}\right) \quad (1.4.45)$$

The contravariant velocities are below

$$U = \xi_x u + \xi_y v + \xi_z w \quad (1.4.46)$$
$$W = \zeta_x u + \zeta_y v + \zeta_z w \quad (1.4.47)$$

and the contravariant velocity U is zero to enforce the boundary condition. The gradient of pressure at the surface can be used to extrapolate accurately to the surface using the interior values. Note that the grid need not be orthogonal to the surface. The derivative of pressure requires the evaluation of metric terms and also state variables along the boundary. Generally, these additional state variables used to determine the gradient are obtained by extrapolation. For example, the density and total magnitude of velocity can be determined by extrapolating entropy and enthalpy from the interior. The direction of the velocity in the plane of the surface must also be extrapolated in three dimensions.

The normal momentum equation approach generally requires more operations than the extrapolation approaches. It is more difficult to implement implicitly in a general-purpose code since both normal and tangential derivatives are involved at the boundary. For example, an implicit treatment of the normal momentum equation leads to a tridiagonal system of equation to be solved at the boundary in two dimensions. For this reason, these equations are usually solved explicitly.

Kutta Condition

It is well known from exact solutions to the potential (incompressible and inviscid) equations that the circulation or, equivalently, the lift must be set in addition to enforcing flow tangency on the surface.[22] At a sharp or a cusped trailing edge, the circulation is usually set through the enforcement of a condition which avoids the occurrence of infinite velocities in the solution. This condition, known as the Kutta or Kutta-Joukowski condition, sets the overall lift in good agreement with experimental observations;[22] generally, the inviscid lift is slightly greater than experiment because of the decambering of the airfoil associated with boundary-layer displacement effects. For a smooth body such as a cylinder, the circulation has to be specified *a priori*.

For numerical solutions to the Euler equations for sharp-edge geometries, the Kutta condition is usually not set directly in the method. Rather, the dissipation inherent in the numerical scheme precludes the occurrence of infinite velocities in the solution. The resulting lift values agree well with compressible potential solutions which enforce the Kutta condition directly, usually through the specification of tangent flow to the trailing-edge bisector angle at the trailing edge. For subcritical flows, a single value of pressure at a sharp trailing edge results from the streamlines along both the lower and upper surfaces being brought to stagnation. For flows which have incurred a loss of total pressure (e.g., through a shock), the local structure of the Euler solutions downstream of the trailing edge corresponds to a slip line, across which the pressure is continuous and the velocity is discontinuous. The velocity stagnates on only one side at the trailing edge; equal pressure at the trailing edge is attained through stagnation of the surface

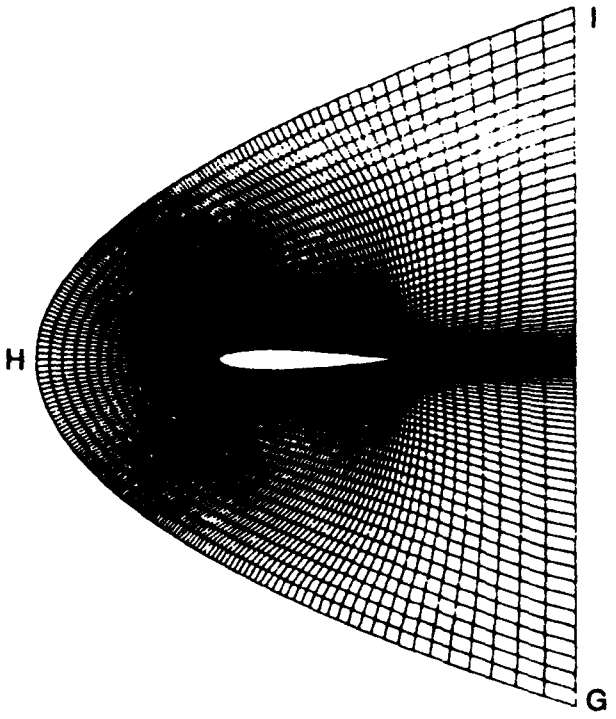
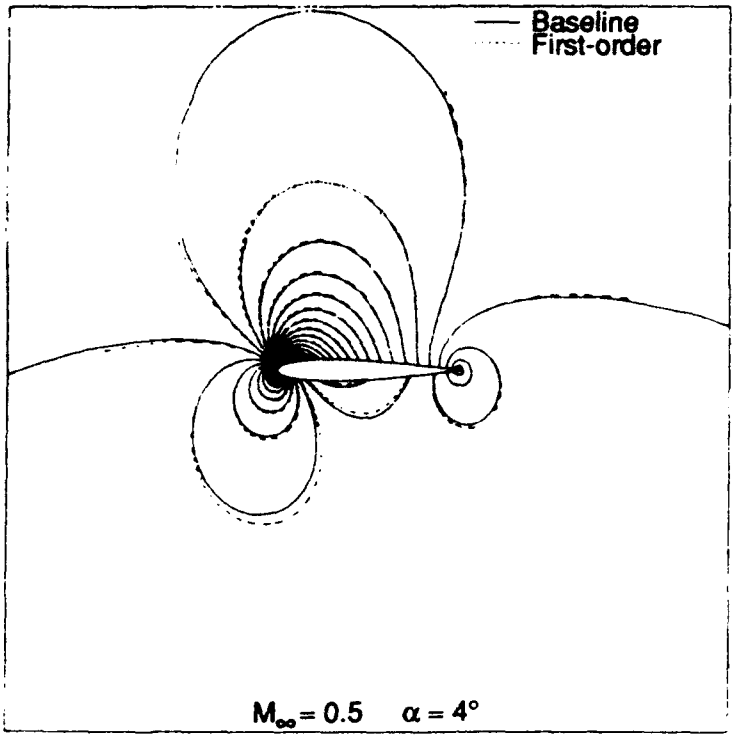


Figure 1.4.4 Computational grid of reduced size for subcritical Euler computations using zero- and first-order far-field boundary conditions.



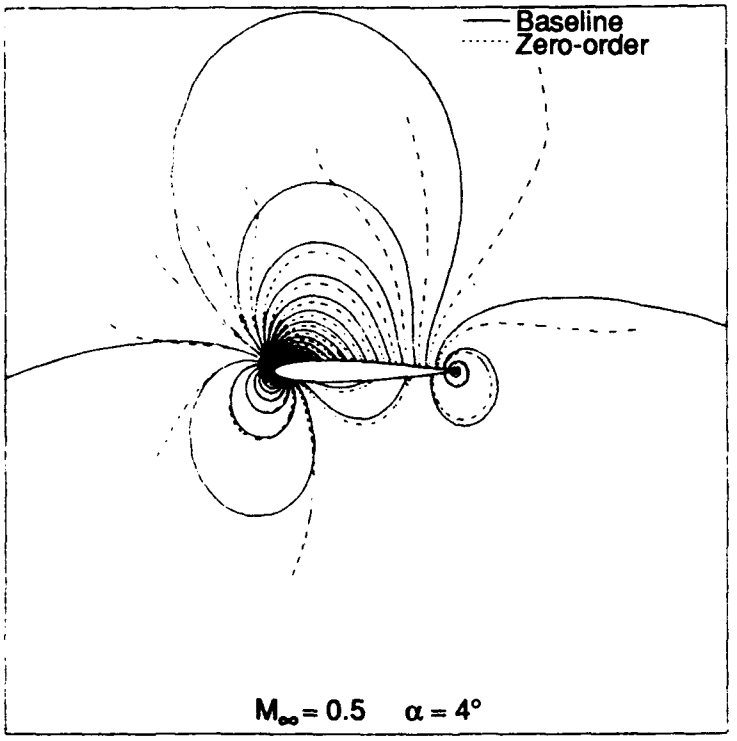


Figure 1.4.5. Pressure contours for NACA 0012 airfoil using zero- and first-order boundary conditions on baseline grid and reduced-extent grid of Fig. 1.4.4 at $M_\infty = 0.5, \alpha = 4$ deg.

streamlines associated with the higher stagnation loss. On the surface of lower stagnation loss, the velocities remain nonzero and tangent to the local surface orientation. Stagnation on both surfaces is physically impossible in this case since the total pressures are different along upper and lower surface streamlines. This structure occurs in a local region of the trailing edge, and it is difficult to resolve the resulting slip line in numerical computations.

The flow separation from sharp leading edges occurring in the subsonic cross-flow plane of swept wings at high angles of attack is also associated with a Kutta condition which enforces smooth flow from the surface. A number of such results are presented in the chapter on Applications and were obtained with the Kutta condition enforced by the dissipation inherent in the numerical scheme. For flows over smooth surfaces, the lift is set either by flow separation (a viscous effect) or through specification of the circulation in the far field; the results of Pulliam presented in the chapter on Applications pose a currently unresolved challenge to state-of-the-art schemes in this respect.

Transpiration and Displacement Effects

For simulating viscous effects in Euler codes, the surface can be moved so that the flow is tangent to an artificial surface defined by projecting the boundary layer displacement thickness distribution normal to the surface.[23,24] Equivalently, the surface boundary condition can be modified to specify a normal velocity. Both methods are used and give equivalent accuracy for representation of viscous effects, although the latter formulation is somewhat simpler since the grid need not be moved to simulate the boundary layer interaction. Raj[25] used the transpiration model to effectively model the effect of variable flap deflections during the development phase of an advanced aircraft program.

1.4.5 Propulsion Simulations

In many applications, it is not necessary to simulate the full details of the propulsion system. Rather, the propulsion system can be considered as a "black box" across or through which the solution can change in a manner consistent with that of a complete modeling. The additional energy and/or swirl added by a jet engine can be specified at a location which might represent a faired-over representation of the actual geometry as a function of engine thrust, for example, and thus avoid the considerable cost and complexity associated with a full simulation. The inflow and outflow conditions appropriate to an actuator disk model are given below.

Outflow

Modeling the downstream end of an engine or propeller corresponds to an outflow case, since the computational domain is downstream and certain boundary conditions need to be specified at a given location (generally on a portion of an axial-normal or slanted plane). For a supersonic outflow, all of the characteristics which reach the actuator disk at the new time level originate from the engine side and, thus, all of the quantities can be specified. The total temperature, total pressure, nozzle pressure ratio, and directions of velocity can be specified, corresponding to a complete reconstruction of the pressure, density, and velocity profiles.

For a subsonic outflow condition, one characteristic reaches the actuator disk at the new time level from the downstream side, corresponding to the computational domain. Thus, one variable needs to be extrapolated from the computation exterior to the disk and the remaining four have to be specified. Generally, the total enthalpy and total pressure of the engine is specified, as well as the two components of velocity. There is some flexibility in the selection of the variable to be extrapolated from the computational domain. For instance, pressure, density, or magnitude of velocity are all valid choices. Taking $|\mathbf{v}|$ as the extrapolated variable, the temperature of the jet can be determined by matching total enthalpy as

$$T = T_t - \frac{\gamma - 1}{2\gamma R}|\mathbf{v}|^2 \tag{1.4.48}$$

The pressure can be determined by assuming an isentropic expansion to stagnation conditions as

$$p = p_t\left(\frac{T}{T_t}\right)^{\frac{\gamma}{\gamma-1}} \tag{1.4.49}$$

and then density can be recovered through the equations of state $\rho = p/(RT)$. The three velocity components can be determined from the velocity magnitude and the imposed two directions of velocity. The boundary conditions in this case are similar to those required at the upstream end of a wind-tunnel simulation, in which the total conditions and velocity directions are prescribed and one variable is extrapolated from the interior.

Inflow

The upstream end of an actuator disk generally corresponds to an inflow case, in which the interior of the computational domain is exterior to the disk. For supersonic inflow, all of the variables should be extrapolated from the interior of the computational domain. Thus, all of the mass which impinges on the disk is swallowed by the device. For subsonic inflow, one characteristic reaches the disk at the new time level from the interior of the modeled system. Thus, four quantities can be extrapolated from the computational domain and one variable specified. The pressure and all velocity components can be extrapolated from the computational domain interior. Since the mass flow should be conserved, an attractive boundary condition option[26] is to specify the mass flow at inflow $\dot{m}$ to provide a means of specifying the density as

$$\rho = \dot{m}A_{inlet}/(\mathbf{v}\cdot\mathbf{n}A) \tag{1.4.50}$$

Here, A is the local surface area of cells that abut the disk, $\mathbf{v}\cdot\mathbf{n}$ is the velocity normal to the disk extrapolated from the interior of the computational domain, and

$$A_{inlet} \equiv \sum A \tag{1.4.51}$$

where the summation extends over those cell areas that abut the inlet disk area.

1.4.6 References

1. MORETTI, G., "A Physical Approach to the Numerical Treatment of Boundaries in Gas Dynamics," Numerical Boundary Condition Procedures, NASA CP-2201, Oct. 1981, pp. 73–95.

2. ROE, P. L., "Upwind Schemes Using Various Formulations of the Euler Equations," Numerical Methods for the Euler Equations of Fluid Dynamics, SIAM, 1983, pp. 14–31.

3. THOMPSON, K. W., "Time Dependent Boundary Conditions for Hyperbolic Systems," Journal of Computational Physics, Vol. 68, 1987, pp. 1–24.

4. JANUS, M. J., "The Development of a Three-Dimensional Split Flux Vector Euler Solver with Dynamic Grid Applications," M.S. Thesis, Department of Aerospace Engineering, Mississippi State University, Aug. 1984.

5. HIRSCH, C., *Numerical Computation of Internal and External Flows, Volume 2, Computational Methods for Inviscid and Viscous Flows*, John Wiley and Sons, 1990.

6. PULLIAM, T. H., "Characteristic Boundary Conditions for the Euler Equations," Numerical Boundary Condition Procedures, NASA CP-2201, Oct. 1981, pp. 165–181.

7. GUSTAFSSON, B., "The Convergence Rate for Difference Approximations to Mixed Initial Value Problems," Mathematics of Computation, Vol. 29, 1975, pp. 396–406.

8. GUSTAFSSON, B., KREISS, H. O. and SUNDSTROM, A., "Stability Theory of Difference Approximations for Mixed Initial Boundary Value Problems, II," Mathematics of Computation, Vol. 26, 1972, pp. 649–686.

9. ROE, P. L., "Remote Boundary Conditions for Unsteady Multidimensional Aerodynamic Applications," ICASE Report No. 86–75 (NASA Contractor Report 178211), Nov. 1986.

10. CHAKRAVARTHY, S. R., "Euler Equations—Implicit Schemes and Boundary Conditions," AIAA Journal, Vol. 21, 1983, pp. 699–706.

11. HEDSTROM, G. W., "Nonreflecting Boundary Conditions for Nonlinear Hyperbolic Systems," Journal of Computational Physics, Vol. 30, 1979, pp. 222–237.

12. RUDY, D. H. and STRIKWERDA, J. C., "A Non-Reflecting Outflow Boundary Condition for Subsonic Navier-Stokes Calculations," Journal of Computational Physics, Vol. 36, 1980, pp. 55–70.

13. GILES, M. B., "Nonreflecting Boundary Conditions for Euler Equation Calculations," AIAA 89–1942, 1989.

14. SAXER, A. and GILES, M. B., "Quasi-3–D Non-Reflecting Boundary Conditions for Euler Equation Calculations," AIAA 91–1603–CP, June 1991

15. ATKINS, H. L. and CASPER, J., "Nonreflective Boundary Conditions for High-Order Methods," AIAA 93–0152, Jan. 1993.

16. VERHOFF, A., STOOKESBERRY, D. and AGRAWAL, S., "Far-Field Computational Boundary Conditions for Two-Dimensional External Flow Problems," AIAA Journal, Vol. 30, No. 11, Nov. 1992, pp. 2585–2594.

17. HIRSCH, C. and VERHOFF, A., "Far-Field Numerical Boundary Conditions for Internal and Cascade Problems," AIAA 89–1943, June 1989.

18. THOMAS, J. L. and SALAS, M. D., "Far-Field Boundary Conditions for Transonic Lifting Solutions to the Euler Equations," AIAA Journal, Vol. 24, No. 7, July 1986, pp 1074–1080.

19. MURMAN, E. M. and COLE, J. D., "Calculation of Plane Steady Transonic Flow," AIAA Journal, Vol. 9, Jan. 1971, pp. 114–121.

20. KLUNKER, E. B., "Contribution to Methods for Calculating the Flow About Thin Lifting Wings at Transonic Speeds—Analytic Expressions for the Far-Field," NASA TN D-6530, 1971.

21. RIZZI, A. W., "Numerical Implementation of Solid-Body Boundary Conditions for the Euler Equations," ZAMM, Vol. 58, 1978, pp. 301–304.

22. THWAITES, B., *Incompressible Aerodynamics*, Oxford at the Clarendon Press, 1960.

23. LIGHTHILL, M. J., "On Displacement Thickness," Journal of Fluid Mechanics, Vol. 4, Part 4, Aug. 1958, pp. 383–392.

24. WHITFIELD, D. L., THOMAS, J. L., JAMESON, A. and SCHMIDT, W., "Computation of Transonic Viscous-Inviscid Interaction Flow," Second Symposium on Numerical and Physical Aspects of Aerodynamic Flows, Springer-Verlag, New York, 1983.

25. RAJ, P. and HARRIS, B., "Using Surface Transpiration with an Euler Method for Cost-Effective Aerodynamic Analyses," AIAA 93–3506, Aug. 1993.

26. HARTWICH, P. M. and FRINK, N. T., "Estimation of Propulsion-Induced Effects on Transonic Flows Over a Hypersonic Configuration," AIAA 92–0523, Jan. 1992.

1.5 EXACT SOLUTIONS

Knowledge of exact solutions to the Euler equations is invaluable in evaluating numerical accuracy and provides an increased understanding of the mathematical and physical nature of inviscid flow. Before recent advances in numerical techniques and an increase in computational capabilities, much theoretical work was conducted to obtain analytic solutions for inviscid flow. This work combines to form the fundamental basis for our understanding of much of gas dynamics. In particular, the solutions of primary importance in this category include shock waves, expansion fans, and contact discontinuities.

However, in spite of several successes at obtaining analytical solutions such as those mentioned above, a general technique for obtaining closed-form solutions is not available. With few exceptions, further simplifying assumptions must be made in order for the governing equations to be solvable in closed form. Therefore, much of the theoretical work is centered around the potential flow equations, which assume irrotational and isentropic flow. Despite the limitations of these assumptions, potential flow theory has provided many reference solutions, as well as valuable insight into the character of both incompressible and compressible flows.

1.5.1 Similarity

Before numerical solution of compressible flows became prevalent, theoretical work concentrated on extending incompressible solutions to represent compressible ones and to relate flows at a given Mach number to those at another Mach number. These techniques rely largely on simplified forms of the Euler equations. By first assuming isentropic and irrotational flow, the governing equations can be expressed in terms of a velocity potential as[1]

$$\begin{aligned}\left(1-\frac{u^2}{a^2}\right)\phi_{xx}-2\frac{uv}{a^2}\phi_{xy}+\\ \left(1-\frac{v^2}{a^2}\right)\phi_{yy}-2\frac{vw}{a^2}\phi_{yz}+\\ \left(1-\frac{w^2}{a^2}\right)\phi_{zz}-2\frac{wu}{a^2}\phi_{zx}=0\end{aligned} \tag{1.5.1}$$

where the velocities in the x, y, and z directions are given in terms of the velocity potential as

$$\begin{aligned}u&=\phi_x\\ v&=\phi_y\\ w&=\phi_z\end{aligned} \tag{1.5.2}$$

Equation (1.5.1) represents a nonlinear partial differential equation for ϕ in terms of x, y, and z.

Unfortunately, because this equation remains nonlinear, analytic solutions of this equation are still not available without further simplifications. For example, invoking the assumption of incompressible flow ($a \to \infty$) immediately yields Laplace's equation: a linear partial differential equation with well-established solution techniques such as complex variables. Also, because the resulting equation is linear, many solutions can be obtained by superposition of other known solutions. Examples of exact solutions that reflect the incompressible assumption include source/sink flows, vortices, and doublets. These fundamental solutions can be combined to obtain solutions over simple configurations such as circles and other elliptically shaped bodies both with and without circulation. In addition to the wealth of knowledge available to analytically solve Laplace's equation for simple configurations, many numerical techniques also exist for obtaining solutions over complicated shapes; the predominant methods are panel methods that are based on Green's function solutions.

Because of the advantages of solving Laplace's equation, it is natural to seek other assumptions that will further simplify Eq. (1.5.1) to make it amenable to solution. One method of achieving this goal is to assume that the flow is perturbed only slightly from the free-stream. With this assumption, the velocity potential in Eq. (1.5.1) can be written as a free-stream component plus a perturbation:

$$\phi(x,y,z)=xV_\infty+\phi'(x,y,z) \tag{1.5.3}$$

After substitution into Eq. (1.5.1) certain terms are discarded based on an order-of-magnitude analysis, and Eq. (1.5.1) can be written as[2]

$$\begin{aligned}(1-M_\infty^2)\frac{\partial^2\phi'}{\partial x^2}+\frac{\partial^2\phi'}{\partial y^2}+\cdots=\\ \frac{M_\infty^2(\gamma+1)}{V_\infty}\frac{\partial\phi'}{\partial x}\frac{\partial^2\phi}{\partial x^2}\end{aligned} \tag{1.5.4}$$

This is the so-called small-perturbation equation, which is valid for subsonic, supersonic, and transonic flow.

If the free-stream flow is subsonic and not too close to Mach 1, Eq. (1.5.4) can be further simplified to

$$(1-M_\infty^2)\frac{\partial^2\phi'}{\partial x^2}+\frac{\partial^2\phi'}{\partial y^2}+\frac{\partial^2\phi'}{\partial z^2}=0 \tag{1.5.5}$$

This can be expressed again in the form of Laplace's equation by applying an affine transformation to this equation. This leads to scaling laws such as the Prandtl-Glauert and Gothert rules, which allow the subsonic compressible flow past a certain profile to be related to the incompressible flow past a second affinely related profile.

If, on the other hand, the flow is purely supersonic, Eq. (1.5.5) can be written as

$$(M_\infty^2-1)\frac{\partial^2\phi'}{\partial x^2}-\frac{\partial^2\phi'}{\partial y^2}-\frac{\partial^2\phi'}{\partial z^2}=0 \tag{1.5.6}$$

which is a hyperbolic, second-order, linear partial differential equation. Although this equation can not be reduced to Laplace's equation, it is, nevertheless, a linear equation and can be solved using linear techniques.

Transonic similarity laws have also been obtained by Guderley[3] in 1946 and by Von Karman[4] and Oswatitsch[5] in 1947. Extensive review of similarity laws for compressible flow can be found in Refs. 6 and 7 as well as in several textbooks such as Refs. 2 and 8Considering two-dimensional flow, through a transformation of the form

$$\phi'(x,y)=A\frac{V_1}{V_2}\Phi'(\xi,\eta) \tag{1.5.7}$$

the parameters M_∞, γ, and the thickness parameter τ can be combined into a single transonic similarity parameter κ

$$\kappa=\frac{1-M_\infty^2}{[\tau M_\infty^2(\gamma+1)]^{2/3}} \tag{1.5.8}$$

Similar solutions are obtained by keeping this parameter invariant regardless of Mach number, thickness parameter, or ratio of specific heats. Afterward, the lift and pressure coefficients can be corrected according to

$$C_{l_2} = AC_{l_1}$$
$$C_{p_2} = AC_{p_1} \tag{1.5.9}$$

where

$$A = \frac{\gamma_1 + 1}{\gamma_2 + 1}\frac{M_1^2}{M_2^2}\frac{1 - M_2^2}{1 - M_1^2} \tag{1.5.10}$$

For example, if the flow over a specified airfoil is known for air ($\gamma = 1.4$) at a Mach number of 0.8, then by matching the similarity parameter, it is found that this solution corresponds to one at a Mach number of 0.814 in different gas corresponding to $\gamma = 1.1$. Although the preceding discussion is for two-dimensional flow, three-dimensional scaling laws exist as well, but require modifications to the aspect ratio to maintain similarity.[9]

1.5.2 Hodograph Solutions

For obtaining analytical solutions for purely supersonic flows, or for subsonic flows not too close to a unity Mach number, the linearized potential flow Eqs. (1.5.5) and (1.5.6) can be utilized as previously mentioned. However, for flows with mixed regions of subsonic and supersonic flow, one must resort to using Eq. (1.5.4) which is not generally solvable in closed form. However, one method that has been effective for yielding exact two-dimensional solutions is the hodograph method,[10] which transforms the stream-function form of the transonic small-disturbance equation into a linear partial differential equation by changing the dependent variables from the spatial coordinates x and y to the flow speed V and the flow angle θ as

$$V^2 c_{VV} + V\left(1 + \frac{V^2}{a^2}\right)c_V + \left(1 - \frac{V^2}{a^2}\right)c_{\theta\theta} = 0 \tag{1.5.11}$$

Solutions to this equation are found using standard techniques such as separation of variables. Once a solution has been found for this equation, the physical geometry must be determined for which the solution applies. Although many solutions may not yield physically realistic shapes, several papers present solutions for the hodograph equation for which the corresponding geometries are representative of a flow of interest.[8,11–20] Reference 21 contains extensive theory, as well as application of the hodograph transformation while Ref. 18 presents a useful summary of exact solutions and relevant references.

One of the most widely used hodograph solutions in recent years for validating accuracy of numerical solutions to the Euler equations is that of Ringleb.[18,22] Shown in Fig. 1.5.1, the physical flow corresponds to that through a curved duct. The flow begins subsonically, accelerates to supersonic flow around the "nose" of the duct, and then decelerates to subsonic flow without a shock. As previously mentioned, this flow has been used by various researchers to evaluate the numerical accuracy of Euler solutions by comparison with this exact solution.[23,24]

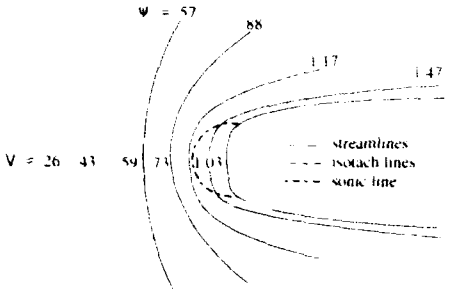


Figure 1.5.1 Ringleb flow

1.5.3 Shockless Airfoils

Although Ringleb flow is clearly a case in which the flow can decelerate from supersonic to subsonic flow without a shock, shock-free transonic flows over airfoils are rare. In the work of Bauer, Garabedian, and Korn,[25] numerical solutions to the hodograph equation have been used to obtain shock-free transonic solutions for airfoils. A further example of a shock-free transonic airfoil is the NLR 7301 airfoil, which has been used as a standard test case for numerical methods for inviscid flow. Note that these solutions are obtained by discretizing the hodograph equation and obtaining a solution numerically. These solutions are, therefore, not exact in that they have not been analytically obtained, but do provide solutions for transonic flow in which isentropic deceleration from supersonic to subsonic flow is present.

1.5.4 Nonunique Solutions

Although nonunique solutions have been known to exist for the potential flow equations,[26,27] until recently, the existence of nonunique solutions for the Euler equations has been largely speculative. In fact, the Euler equations were used in the previous references 26–27 to validate the nonuniqueness of the potential assumption. During these studies, nonunique solutions were sought with the Euler equations; however none were found.

For the potential flow equations, the nonuniqueness has been attributed to the breakdown of the validity of the assumptions inherent in the equation (namely, the assumptions of irrotational and isentropic flow). Because the Euler equations do not make these assumptions, the appearance of nonuniqueness has until recently been somewhat doubtful. However, in 1991, Jameson[28] computed nonunique solutions for four airfoils designed by an optimization method based on control theory.[29]

An example of nonuniqueness for one of the airfoils (denoted

as J-78) is shown in Fig. 1.5.2. This airfoil was designed

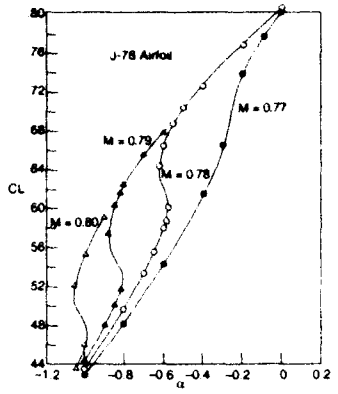


Figure 1.5.2 Examples of nonunique solutions to Euler equations.

for a Mach number of 0.78 and a lift coefficient of 0.6. Figure 1.5.2 clearly shows the nonunique behavior of this airfoil as evidenced by the lift coefficient exhibiting multiple values at a given angle of attack. As mentioned in Ref. 28, this nonuniqueness persists even on very fine grids and provides strong evidence that Euler solutions for airfoils are not necessarily unique.

1.5.5 Exact Solutions for Supersonic Flows

For supersonic flows, several exact solutions exist. Particularly, for flows without shocks, the equations are isentropic and irrotational and lead to solutions such as Prandtl-Meyer expansions. However, even for flows with shock waves, many solutions exists for simple wave flows such as wedges, cones, converging-diverging ducts, diamond-shaped airfoils, and flows around blunt bodies. Techniques for solving these problems are discussed in many textbooks on gas dynamics and aerodynamics.[30,31,32]

1.5.6 Riemann Problem

One of the most important exact solutions, which has proved very useful in designing numerical schemes, is Riemann's initial value problem.[32] This is a generalization of a shock tube problem in which initial conditions are specified across a diaphragm, but the velocity on either side of the diaphragm may be nonzero. At $t = 0$, the diaphragm is instantaneously broken, and the evolution of the flow-field is tracked. At a given time, the general solution consists of a shock wave and a contact discontinuity traveling in one direction with a speed of $u+a$ and u, respectively, while an expansion fan is traveling in the opposite direction at speed $u - a$. (See Fig. 1.5.3.)

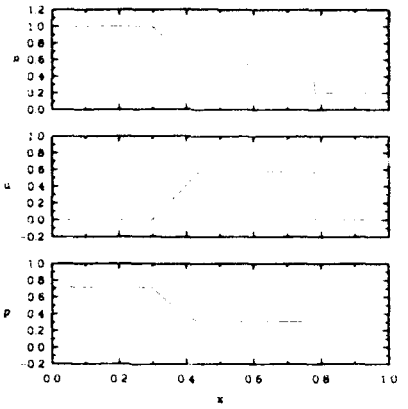


Figure 1.5.3 Riemann problem

In Fig. 1.5.3, a shock wave is traveling to the right which results in a sharp increase in the density, velocity, and pressure as it passes. The expansion is traveling to the left and induces a more gradual increase in the velocity of the fluid that was originally to the left of the diaphragm. The contact, also traveling to the right, induces a jump in the density, but has no effect on the pressure and velocity.

1.5.7 References

1 SHAPIRO, A., *The Dynamics and Thermodynamics of Compressible Fluid Flow*, vol. I. John Wiley & Sons, 1953.

2 LIEPMANN, H., and ROSHKO, A., *Elements of Gasdynamics*. John Wiley & Sons, 1957.

3 GUDERLEY, G., "Die Urache für das Auftreten von Verdichtungsstößen in Gemischten Unter- and Übertschallströmungen," Völkenrode Repts. and Transl., no. 110, 1946.

4 T. VON KARMAN, "The Similarity Law of Transonic Flow," J. Math. Phys., vol. 182, no. 26, 1947.

5 OSWATITSCH, K., "A New Similarity Law for Profiles Valid in the Transonic Region," Roy. Aircraft Estab. TN Aero 1902, 1947.

6 ZIEREP, J., *Similarity Laws and Modeling*. New York: Marcel Dekker, Inc., 1971.

7 OSWATITSCH, K., "Similarity and Equivalence in Compressible Flow," in *Advances in Applied Mechanics* (Dryden, H. L., and VON KARMAN, T., eds.), vol. VI, Academic Press, 1960.

8 SHAPIRO, A., *The Dynamics and Thermodynamics of Compressible Fluid Flow*, vol. II. Krieger Publishing Company, 1982.

9 SPREITER, J., "On the Application of Transonic Similarity Rules," Tech. Rep. Tech. Note 2726, NACA, 1952.

10 LIGHTHILL, M. J., "The Hodograph Transformation," in *Modern Developments in Fluid Dynamics, High Speed Flow*, vol. 1, Clarendon Press, 1953.

11 **SOBIECZKY, H.**, *Application of Generalized Potentials on Plane Transonic Flow, IUTAM Symposium Transsonicum II, Göttingen*. Springer-Verlag Berlin, Heidelberg, New York, 1975.

12 **EBERLE, A.**, *An Exact Hodograph Method for the Design of Supercritical Wing Sections, IUTAM Symposium Transsonicum II, Göttingen*. Springer-Verlag Berlin, Heidelberg, New York, 1975.

13 **BOERSTOEL, J. W.**, *Review of the Application of Hodograph Theory to Transonic Airfoil Design and Theoretical and Experimental Analysis of Shock-free Airfoils, IUTAM Symposium Transsonicum II, Göttingen*. Springer-Verlag Berlin, Heidelberg, New York, 1975.

14 BOERSTOEL, J. W., "A Transonic Hodograph Theory for Airfoil Design," in *IMA Conf. Comp. Methods and Problems in Aeron. Fluid Dynamics*, Manchester, 1974. Also NLR Report MP74024 U.

15 BOERSTOEL, J. W., and HUIZING, G. H., "Transonic Shock-free Aerofoil Design by an Analytic Hodograph Method," AIAA 74–539, 1974.

16 NIEUWLAND, G. Y., and SPEE, B. M., "Transonic Airfoils: Recent Developments in Theory, Experiment, and Design," Annual Review of Fluid Mechanics, vol. 5, pp. 119–150, 1973.

17 VERHOFF, A., STOOKESBERRY, A., and MICHAL, T., "Hodograph Solution for Compressible Flow Past a Corner and Comparison with Euler Numerical Predictions," AIAA 91–1547, 1991.

18 CHIOCCHIA, G., "Exact Solution to Transonic and Supersonic Flows," Test Cases for Inviscid Flow Field Methods, AGARD Advisory Report AR-211, May 1985.

19 GUDERLEY, G., and YOSHIHARA, H., "The Flow over a Wedge Profile at Mach Number 1," J. Aeronautical Sciences, vol. 17, no. 11, 1950.

20 CHATTOT, J., "Symmetrical Flow Past a Double Wedge at High Subsonic Mach Numbers," J. Fluid Mech., vol. 86, pp. 161–177, 1978.

21 GUDERLEY, K. G., *The Theory of Transonic Flow*. Pergamon Press, 1962.

22 RINGLEB, F., "Exakte Losungen der Differentialgleichungen Einer Adiabatischen Gasstromung," ZAMM, vol. 20, 1940.

23 BARTH, T., and FREDERICKSON, P., "Higher Order Solution of the Euler Equations on Unstructured Grids Using Quadratic Reconstruction," AIAA 90–0013, Jan. 1990.

24 HALT, D., and AGARWAL, R., "A Compact Higher Order Characteristic Based Euler Solver for Unstructured Grids," AIAA 91-3234, Sept. 1991.

25 BAUER, F., GARABEDIAN, P., and KORN, D., *Supercritical Wing Sections*. Springer-Verlag, 1975.

26 SALAS, M. D., and GUMBERT, C. R., "Breakdown of the Conservative Potential Equation," Tech. Rep. TP 2539, NASA, 1986.

27 SALAS, M. D., JAMESON, A., and MELNIK, R. E., "A Comparative Study of the Nonuniqueness Problem of the Potential Equation," Tech. Rep. TP 2385, NASA, 1985.

28 JAMESON, A., "Airfoils Admitting Non-Unique Solutions of the Euler Equations," AIAA 91–1625, June 1991.

29 JAMESON, A., "Aerodynamic Design via Control Theory," Tech. Rep. 8864, ICASE, Nov. 1988.

30 KUETHE, A., and CHOW, C., *Foundations of Aerodynamics*. John Wiley & Sons, 1976.

31 BERTIN, J., and SMITH, M., *Aerodynamics for Engineers*. Prentice-Hall, 1979.

32 COURANT, R., and FRIEDRICHS, K., *Supersonic Flow and Shock Waves*. Springer-Verlag, 1976.

This page has been deliberately left blank.

Chapter 2

Numerical Schemes and Algorithms

2.1 DISCRETIZATION TECHNIQUES

In order to describe the numerical methods currently used for solving the system of Euler equations, we have to begin with a brief presentation of the discretization techniques which allow to transform the continuous problem into a system of discrete equations to be solved on a computer.

Three main steps can be considered. First, the space-time discretization concerning the independent variables (this is the mesh generation problem), then the choice of a discrete representation of the flow variables (approximation of dependent variables) and thirdly the derivation of a set of discrete equations linking the flow variables on the grid in space and time (definition of a numerical scheme).

Grid generation

The mesh generation techniques have become a field of research and expertise which the quality of numerical results greatly depends on. This important subject is treated in detail in Section 2.2, but it is necessary to introduce here a few ideas for presenting the following steps of equation discretization.

At first, the space-time domain where the problem is to be solved must be discretized. Except for very special situations, the time and the space dimensions are treated independently and however complex could be the space discretization, the dependent variables are all represented at same time values.

The space discretization process consists in replacing the continuous three-dimensional domain where the flow is studied by a mesh or a grid made of points or nodes connected by edges and faces which bound cells or elements. The union of the cells forms a partition of the whole computational domain.

The common practice in mesh generation is to use either triangular or quadrilateral cells in two dimensions (tetrahedral or hexahedral in three dimensions).

The overall arrangement of these cells may be either "structured or unstructured".

The structured grids are made of families of grid lines in two dimensions (2D) and of families of grid surfaces in three dimensions (3D). These grid lines or grid surfaces are indexed by integers so that each node at their intersection is indexed by a set of indices. The cells with the nodes as vertices can be indexed in the same way. Connectivity rules are identical for all cells so that we can invoke a "stencil".

In contrast, the unstructured grids consist of an arbitrary assembly of cells with only the possibility to index each one by a single integer and no regular pattern or relationship exists between cell and node numbering. The data structure management necessitates the definition and the storage of pointers and index tables.

Besides the nodes previously described as vertices of the cells, it can be useful to consider other points in the grid where discrete dependent variables are defined. We mean for example mid-points of edges and centroids of faces or centroids of primary cells. It is natural to use these auxiliary points to build secondary cells and this will be illustrated in the subsection 2.1.2 about finite volume methods where the distinction between cell-centered and cell-vertex schemes is made.

Approximation of dependent variables

After the discretization of the space and time independent variables we can proceed to the discrete representation of the dependent or flow variables. The crudest and the simplest discrete representation of a scalar function of several independent variables is limited to setting up the values of this function at the grid points without concern of its value elsewhere. This idea is the basis for the finite difference approach which is almost always applied with structured 2D quadrilateral or 3D hexahedral grids of either cartesian or more often body-fitted curvilinear type.

The alternative for the discrete approximation of functions on a grid is to consider piecewise polynomials locally defined on each cell by a small number of values or "degrees of fredom". This topic is covered by the theory of approximation and interpolation in many mathematical textbooks and is particularly important for the design of finite element methods and of spectral methods.

The pointwise approximation used for finite differences could be included in the general theory of approximation by resorting to the Dirac measure but that is of little practical interest.

In the class of piecewise polynomial approximations, the simplest one is the piecewise constant approximation and it is the basis for the finite volume methods described in subsection 2.1.2.

Piecewise linear approximation is generally associated with triangular or tetrahedral meshes, piecewise bilinear with quadrilateral meshes and trilinear with hexahedral meshes. The approximate functions are either continuous (most frequently in finite element methods) or discontinuous (in finite volume methods).

Higher order polynomials also deserve some attention. For instance, piecewise parabolic interpolation has been the specific device attached to the PPM method proposed by Woodward and Collela[1]. More recently, high order polynomial interpolation has been included in the "reconstruction step" for Essentially Non Oscillatory (ENO) schemes developed by Harten[2] and Osher[3].

Derivation of the discrete equations

We now reach the third stage in the discretization process, namely the derivation of the discrete equations linking flow variables on the grid at different time steps. Three distinct routes may be taken according to the choice of a formulation representing the system of Euler equations.

1.- *Finite Difference Methods:*
Starting from one system of first order partial differential equations (in conservation form or not), time and space derivatives are replaced by finite differences resulting from the application of Taylor series at grid points.

2.- *Finite Volume Methods:*
The integral form of the Euler equations given by Eq. (1.3.1) and corresponding to the laws of conservation for mass, momentum and energy are applied on each cell with the assumption of a constant value for the conservative quantities $(\rho,\rho\vec{u},\rho E)$ whereas the definition of the fluxes on the cell boundary allows a considerable number of variants for controlling the accuracy, the robustness and the efficiency of these methods. The only mandatory rule is that the flux evaluation on an interface between two cells be the same for ensuring the conservation property.

3.- *Finite Element Methods:*
The weak formulation of Euler equations, such as the one given by Eq. (1.3.13) is applied after the choice of the spaces of approximation both for the dependent variables and the test functions. Triangles in 2D and tetrahedra in 3D are systematically chosen by practitionners of finite element methods. However, the name of finite element method should be reserved to the case where test functions are at least polynomial of degree one, since the use of a test function which is constant by cell boils down to the finite volume approach.

At this point it is interesting to notice that the three previous methods for deriving discrete equations can lead to very similar results and it is often possible to show the equivalence of some finite difference and finite volume schemes on structured grids and of some finite volume and finite element methods on unstructured triangular grids. Other discretization methods exist but are only marginally used compared to the three previously mentioned methods. Namely they are the spectral methods and fluctuation splitting approaches. the first one is somewhat related to finite element discretization since it uses continuous basis functions which are generally some high order polynomials but of non local support, in contrast with the case of finite elements. Some link can be found also with finite differences by the fact that the grid of collocation points is of structured type for two- or three-dimensional problems. The continuous approximation is not very suitable for solution of Euler equations with discontinuities without resorting to complicated shock-fitting or shock-tracking and it seems that no production code exists based on spectral methods. The fluctuation splitting approach is relatively recent [4] and it can be considered as an attempt to mimic the propagation of characteristic waves in order to replace the rather one-dimensional evaluation of normal fluxes in finite volume methods by a more multidimensional concept.

Time and space discretization

What has been said above concerns more the space discretization than the way the time variable influences the design of a numerical method. Let us assume first that we are considering a time-dependent problem where the time evolution of the discrete solution must be represented accurately. It seems natural with present day three-dimensional problems to propose that time and space discretization would be made separately. However, with one space dimension equations for which the early finite difference schemes where studied, it was logic to discretize together the time and space derivatives.

The Lax-Friedrichs and Lax-Wendroff schemes which were at the basis of the numerical solution of hyperbolic systems of equations indeed combine time and space discretization (see section 2.4 below). Both are two-level explicit schemes. This is also the case for the family of implicit schemes derived by Lerat as an extension of the Lax-Wendroff schemes. They also combines both time and space discretization in a coupled manner.

By contrast, when space discretization is made first, leading to what is called a semi-discretization, the resulting system of ordinary differential equations in time can be solved by various methods. Two classes of time integration methods widely used for solving Euler equations are the linear multistep methods specially studied and used by Beam and Warming [5] and the Runge Kutta methods mainly developed by Jameson, Schmidt and Turkel [6].

2.1.1 Finite Difference Techniques

As said before, the finite difference methods rely upon the approximation of a derivative by the ratio of two differences according to the definition of this derivative and with an order of accuracy which is estimated by Taylor expansions. The simplest and largest use of finite difference discretization for solving the Euler system of equations actually appears in the approximation of the time derivative.

For a function $u(x,t) = u_i(t)$ its first derivative at time t is defined by:

$$\frac{du_i}{dt} = \lim_{\Delta t \to 0} \frac{u_i(t+\Delta t) - u_i(t)}{\Delta t} \qquad (2.1.1)$$

If we consider a uniform time discretization with $t^n = n\,\Delta t$ and $u_i^{\,n} = u_i(t^n)$, this time derivative can be approximated by the following finite difference formulas:

$$\left[\frac{du_i}{dt}\right]^n = \left[\frac{du_i}{dt}\right]_{t=t^n} = \frac{\Delta u_i^{\,n}}{\Delta t} + O(\Delta t) \qquad (2.1.2)$$

$$\left[\frac{du_i}{dt}\right]^n = \left[\frac{du_i}{dt}\right]_{t=t^n} = \frac{\Delta u_i^{\,n-1}}{\Delta t} + O(\Delta t) \qquad (2.1.3)$$

$$\left[\frac{du_i}{dt}\right]^n = \left[\frac{du_i}{dt}\right]_{t=t^n} = \frac{1}{2}\frac{(\Delta u_i^{\,n} + \Delta u_i^{\,n-1})}{\Delta t} + O(\Delta t^2) \qquad (2.1.4)$$

where $\Delta f^n = f^{n+1} - f^n$.

These formulas are respectively forward, backward and central finite difference approximations at t^n and are first order accurate for the one-sided difference formulas Eq. (2.1.2-3) and second order accurate for the central approximation Eq. (2.1.4) . However one can see that Eq. (2.1.2) is second order accurate with respect to the value of the time derivative at $t = t^{n+1/2} = (n+1/2)\,\Delta t$.

In order to be more precise and to give an example of such finite difference formulas, consider the one dimensional Euler system of equations:

$$\frac{\partial \mathbf{w}}{\partial t} + \frac{\partial \mathbf{F}}{\partial x} = 0 \qquad (2.1.5)$$

It can be discretized in time (semi-discretization) by finite differences according to:

$$(1+\xi)\,\Delta\mathbf{w}^n - \xi\,\Delta\mathbf{w}^{n-1} = -\Delta t\,[\theta\,\frac{\partial \mathbf{F}^{n+1}}{\partial x} + (1-\theta)\,\frac{\partial \mathbf{F}^n}{\partial x}]$$
$$+ (\theta - \xi - 1/2)\,O(\Delta t^2) + O(\Delta t^3)$$

The linearization of the $\mathbf{F}^{n+1}$ term as introduced by Briley and Mc Donald [7] can be used:

$$\mathbf{F}^{n+1} = \mathbf{F}^n + \Delta t\,(\frac{\partial \mathbf{F}}{\partial t})^n + O(\Delta t^2)$$

$$(\frac{\partial \mathbf{F}}{\partial t})^n = \mathbf{A}^n \Delta\mathbf{w}^n + O(\Delta t), \qquad \mathbf{A} = \frac{\partial \mathbf{F}}{\partial \mathbf{w}}$$

giving:

$$(1 + \xi + \Delta t\;\theta\,\frac{\partial}{\partial x}\,\mathbf{A}^n)\,\Delta\mathbf{w}^n =$$

$$-\Delta t\,\frac{\partial \mathbf{F}^n}{\partial x} + \xi\,\Delta\mathbf{w}^{n-1} + (\theta - \xi - 1/2)\,O(\Delta t^2) \qquad (2.1.6)$$

with second order accuracy if $\theta = 1/2 + \xi$.

Explicit ($\theta = 0$) and implicit schemes are obtained after a

space discretization and we get the family of the Beam and Warming schemes [5] if central discretization is used for the space derivatives (see Section 2.4.2 for a discussion of these schemes).

When space and time discretizations are made independently, we can reverse the order of them and begin with the space discretization. For a function $v(x)$, considering a uniform distribution of points $x_i = i\Delta x$, with nodal values $v_i = v(x_i)$, the approximation of the first derivative of v can be chosen according to one of the following finite difference formulas:

$$\left[\frac{dv}{dx}\right]_i = \left[\frac{dv}{dx}\right]_{x=x_i} = \frac{\delta^+ v_i}{\delta x} + O(\delta x);\ \delta^+ v_i = v_{i+1} - v_i \qquad (2.1.7)$$

$$\left[\frac{dv}{dx}\right]_i = \left[\frac{dv}{dx}\right]_{x=x_i} = \frac{\delta^- v_i}{\delta x} + O(\delta x);\ \delta^- v_i = v_i - v_{i-1} \qquad (2.1.8)$$

$$\left[\frac{dv}{dx}\right]_i = \left[\frac{dv}{dx}\right]_{x=x_i} = \frac{\delta v_i}{\delta x} + O(\delta x^2);\ \delta v_i = v_{i+1/2} - v_{i-1/2} \qquad (2.1.9)$$

Direct application of these finite difference formulas to the derivative in Eq. (2.1.5) can be considered with a preference for the third one which is second order accurate giving:

$$\frac{d\mathbf{w}_i}{dt} + (\mathbf{F}_{i+1/2} - \mathbf{F}_{i-1/2}) / \delta x = 0 \qquad (2.1.10)$$

Expressing $\mathbf{F}_{i+1/2}$ in terms of the nodal values $\mathbf{w}_i$ (with a very large number of possibilities) leads to a (non linear) system of ordinary differential equations to be solved by an appropriate time integration such as the linear multistep or the Runge-Kutta methods.

If we restrict ourselves to the case of two- (time) level difference schemes, the most useful formulation of such schemes is:

$$\frac{\Delta \mathbf{w}_i}{\Delta t} + \frac{\delta \mathbf{h}_i}{\delta x} = 0 \qquad (2.1.11)$$

or else:

$$\Delta \mathbf{w}_i = -\sigma(\mathbf{h}_{i+1/2} - \mathbf{h}_{i-1/2}) \ ; \ \sigma = \Delta t/\delta x$$

with a numerical flux:

$$\mathbf{h}_{i+1/2} = \mathbf{h}^{(f)}(\mathbf{w}_{i-l+1},...,\mathbf{w}_{i+l};\Delta \mathbf{w}_{i-l+1},...,\Delta \mathbf{w}_{i+l};\sigma) \qquad (2.1.12)$$

where $\mathbf{h}^{(f)}$ is a continuous function satisfying the consistency condition:

$$\mathbf{h}^{(f)}(\mathbf{w},...,\mathbf{w};0,...,0;\sigma) = \mathbf{F}(\mathbf{w}) \text{ for any } \mathbf{w} \qquad (2.1.13)$$

Eq. (2.1.11) represents a discrete conservation law in a cell $c_i = [(i-1/2)\ \delta x\ ,\ (i+1/2)\ \delta x]$ if we consider $\mathbf{w}_i$ as a local average:

$$\mathbf{w}_i = \frac{1}{\delta x} \int_{c_i} \mathbf{w}(x, n\Delta t)\, dx$$

and $\mathbf{h}_{i+1/2}$ as a time average of the flux accross the cell side $x = (i+1/2)\delta x$.

Schemes written in the form given by Eq.(2.1.11-13) are said to be conservative because summation of Eq.(2.1.11) over a set D of contiguous cells provides, by cancellation of numerical fluxes at interior interfaces, a discrete form of the conservation law on D:

$$D = \bigcup_{i=p,q} C_i = [(p-1/2)\delta x, (q+1/2)\delta x],$$

$$\int_D [\mathbf{w}(x,(n+1)\Delta t) - \mathbf{w}(x,n\Delta t)]dx = -\Delta t(\mathbf{h}_{q+1/2} - \mathbf{h}_{p-1/2})$$

The usefulness of conservative schemes results from their property of capturing discontinuities with the correct levels corresponding to the Rankine-Hugoniot relations. We shall come back on this point when discussing the finite volume discretization.

A very detailed study of finite difference schemes for 1-D hyperbolic problems has been presented by Lerat [8] with emphasis on space-centered approximation. From the large class of schemes discussed one can extract a yet very general subclass of two-level either explicit or linearly implicit schemes involving at most three points at the new time level:

$$\Delta \mathbf{w}_i + 1/2\ \sigma\ \delta[\mathbf{M}\mu(\Delta \mathbf{w})]_i - 1/4\ \delta[\mathbf{P}\delta(\Delta \mathbf{w})]_i$$
$$= -\sigma\ \delta(\mu \mathbf{F})_i + 1/2\ \delta(\mathbf{Q}\delta \mathbf{w})_i \qquad (2.1.14)$$

where $\mu \mathbf{w}_{i+1/2} = 1/2\ (\mathbf{w}_i + \mathbf{w}_{i+1})$, and $\mathbf{M}_{i+1/2}$, $\mathbf{P}_{i+1/2}$ and $\mathbf{Q}_{i+1/2}$ are 3x3 matrices depending on the values of $\mathbf{w}$ at t^n.

This class of schemes contains most of two-level schemes such as first order upwind and TVD schemes, the second order centered schemes of Beam and Warming and the generalization of explicit and implicit Lax-Wendroff schemes proposed by Lerat[9]. A good representative of this family, namely an implicit Lax-Wendroff scheme of second order of accuracy, is given below:

$$\Delta \mathbf{w}_i - 1/2\ \sigma^2\ \delta[(\mu \mathbf{A})^2\ \delta(\Delta \mathbf{w})]_i$$
$$= -\sigma\ \delta(\mu \mathbf{F})_i + 1/2\ \sigma^2\ \delta[(\mu \mathbf{A})\ \delta \mathbf{F}]_i \qquad (2.1.15)$$

Until now, we have considered only one dimensional finite difference schemes. Extension of these schemes to the multidimensional Euler system:

$$\partial \mathbf{w}/\partial t + \partial \mathbf{F}/\partial x + \partial \mathbf{G}/\partial y + \partial \mathbf{H}/\partial z = 0 \qquad (2.1.16)$$

is relatively straightforward if a cartesian grid is used even though the number of possibilities is quickly increasing for the approximation of higher order space derivatives, for the linearization of the $\mathbf{F}$, $\mathbf{G}$ and $\mathbf{H}$ fluxes and for the combination of differentiation and linearization.

For example, direct extension of the explicit Lax-Wendroff scheme can be made either according to the Ni scheme using a predictor at the mid-cell point or according to another scheme with a predictor step centered at the mid-point of interfaces giving better dissipative properties. This explicit scheme was defined to be used in combination with the implicit step consisting of successive application of the left hand side of Eq.(2.1.15) in each direction (see Lerat, Sides [10]).

Use of cartesian grids (without local refinement) is not suitable for computing flows past bodies with curved boundaries presenting too small a radius of curvature or a complex geometry. Boundary conforming curvilinear meshes are much more preferred.

The current procedure for applying finite differencing on curvilinear structured grids is founded upon a mapping from the physical space (x,y,z) to a computational (or reference) space (ξ,η,ζ) equipped with a uniformly spaced cartesian grid. The transformation has to be one - to - one and is assumed to be smooth enough for being differentiable. Then the system of Euler equations to be solved can be formulated in the computational space by the change of independent variables (space coordinates). Special care must be taken in order to keep a conservative formulation with cartesian components of velocity as unknowns (see Viviand[11] and Vinokur[12]). With the introduction of contravariant velocity components as:

$$U = u\,\xi_x + v\,\xi_y + w\,\xi_z$$

$$V = u\,\eta_x + v\,\eta_y + w\,\eta_z$$

$$W = u\zeta_x + v\zeta_y + w\zeta_z$$

the system (2.1.16) can be written as follows:

$$\partial\hat{\mathbf{w}} / \partial t + \partial\hat{\mathbf{F}} / \partial\xi + \partial\hat{\mathbf{G}} / \partial\eta + \partial\hat{\mathbf{H}} / \partial\zeta = 0 \qquad (2.1.17)$$

with: $\hat{\mathbf{w}} = \mathbf{w} / J$, and $J = \partial(\xi,\eta,\zeta) / \partial(x,y,z)$,

$$\hat{\mathbf{F}} = [U\mathbf{w} + p(0, \nabla\xi, 0)^t] / J$$

$$\hat{\mathbf{G}} = [V\mathbf{w} + p(0, \nabla\eta, 0)^t] / J$$

$$\hat{\mathbf{H}} = [W\mathbf{w} + p(0, \nabla\zeta, 0)^t] / J$$

Special care must be exercised during the discretization process in order to satisfy requirements such as the free-stream or uniform flow is an exact solution of the discretized equations. The conservative property of the final finite difference scheme has also to be checked.

This induces a close dependence between the choices of the finite difference formulas for the dependent variables w, and for the independent variables x,y,z with respect to the cartesian coordinates ξ,η,ζ in the computational domain.

The alternative to a discretization in a computational domain after a coordinate transformation is to directly work out the discretization in the physical space. This is the basic principle for the finite volume formulation.

2.1.2 Finite Volume Techniques

As said above, the finite volume approach relies directly on the application of the integral form of balance laws. Assuming that the domain of integration is independent of time, we write these conservation laws for an arbitrary spatial domain Ω bounded by Σ.

$$d(\int_{\Omega} \mathbf{w}\, dx\, dy\, dz) / dt + \int_{\Sigma} (\bar{\mathbf{F}}.\bar{n})\, d\sigma = 0 \qquad (2.1.18)$$

where $\mathbf{w}$ is the vector of conservative variables and $\bar{\mathbf{F}}$ the flux tensor.

If we assume that the physical domain is covered by a collection of elementary polyhedral cells forming a partition of this domain, the application of Eq.(2.1.18) at the level of each elementary cell ensures a conservative discretization, that is to say ensures the validity of Eq.(2.1.18) for any union of elementary cells if the numerical approximation for $(\bar{\mathbf{F}}.\bar{n})$ is unique on the interface between two contiguous cells.

To be more specific, let us consider two neighbouring cells Ω_I and Ω_J, their common interface Σ_{IJ} with its unit normal $\bar{n}$ oriented from Ω_I toward Ω_J. We denote by $\mathbf{w}_I$ the mean value of $\mathbf{w}$ on Ω_I:

$$|\Omega_I|\mathbf{w}_I = \int_{\Omega_I} \mathbf{w} dx dy dz \qquad (2.1.19)$$

and by $\bar{\mathbf{F}}_{IJ}$, the mean value of the normal flux on Σ_{IJ}:

$$|\Sigma_{IJ}|\,\bar{\mathbf{F}}_{IJ} = \int_{\Sigma_{IJ}} \bar{\mathbf{F}}.\bar{n} d\sigma \qquad (2.1.20)$$

If we call N_I the set of indexes of the cells surrounding Ω_I and having an interface in common with it, Eq.(2.1.18) applied to Ω_I becomes:

$$d(\mathbf{w}_I\ |\Omega_I|) / dt + \sum_{J \in N_I} \bar{\mathbf{F}}_{IJ}\ |\Sigma_{IJ}| = 0 \qquad (2.1.21)$$

Considering the set of all cells $\{\Omega_I\}$, we get the discretized equations for the whole physical domain with unknowns $\{\mathbf{w}_I\}$ if $\bar{\mathbf{F}}_{IJ}$ is evaluated from a finite number of cell unknowns such as at least $\mathbf{w}_I$ and $\mathbf{w}_J$ but also from $\mathbf{w}_K$, with K belonging to N_I and N_J and possibly to other cells (see Fig. 2.1.1 in 2D for simplification).

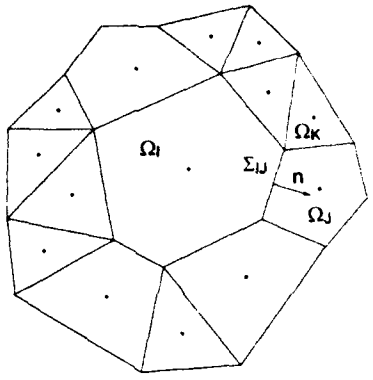


Figure 2.1.1 Control volumes and nodes in a general finite volume method.

The set of equations resulting from Eq.(2.1.21) is clearly a semi-discretization with possibly a large freedom not only for the choice of the grid and for the definition for cells or "control volumes" on which Eq.(2.1.18-21) are applied but also for the choice of the dependency of $\bar{\mathbf{F}}_{IJ}$ with respect to the unknowns $\{\mathbf{w}_I\}$ both in space and in time.

About the first topic, clearly both structured and unstructured grids are concerned by this formulation. In both cases the three-dimensional grid is made of polyhedra where we can distinguish the sets of vertices, edges, faces and cells. Three types of cell arrangements are used in practice: cell vertex, node centered and cell centered methods.

Cell vertex methods

Cell vertex finite volume methods seem to be the oldest ones (McDonald[13] 1971, MacCormack and Paullay[14] 1972) and were devised for the solution of two-dimensional time-dependent Euler equations. They keep strong favour with several variants such as the Ni scheme[15] or its modification by Hall[16].

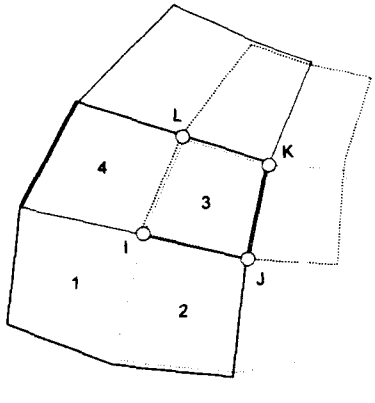


Figure 2.1.2 Cell vertex finite volume method.

In the cell vertex methods the flow variables are assigned to vertices (mesh nodes) and the control volume attached to node I is made of the collection of elementary cells (indexed by the set N_I) surrounding this node. With this choice of control volumes, we see, for example in a two dimensional structured grid (see Fig. 2.1.2), that two neighbouring control volumes have in common one or two mesh cells and that the

union of all the control volumes covers four times the physical domain since each quadrilateral belongs in general to four different control volumes. For an unstructured grid with triangles, each triangle belongs to the three control volumes attached to its three vertices. Any edge belongs to two control volumes and for conservation of physical quantities the fluxes must be evaluated in the same manner for both adjacent control volumes on their interface.

Cell centered methods

known approach consists in taking elementary control volumes. Thus the mean value w_I can be to some centroid of the mesh cell.

For triangles or tetrahedra the centroid is normally the point of intersection of medians and for quadrilaterals and hexahedra it is located at the intersection of straight lines joining the mid-points of opposite sides. There is not much room for geometric variants in the cell centered finite volume schemes. The only dilemma is for evaluating fluxes on the interface between two cells. This evaluation can be either a central interpolation or an upwind formulation.

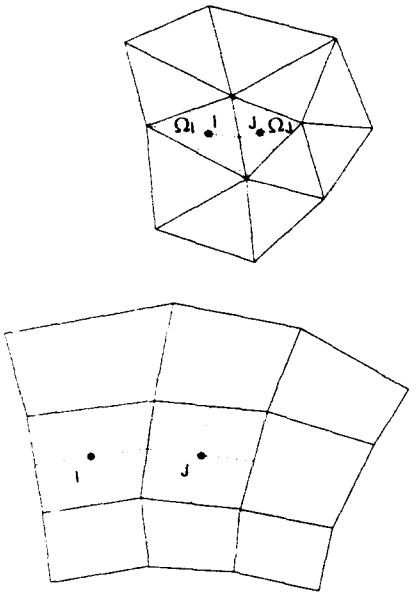


Figure 2.1.3 Cell centered finite volume methods.

The first choice was taken by Jameson et al.[6] and Rizzi[17] with a simple mean value of fluxes from the two adjacent cells, the lack of dissipative properties of the resulting scheme imposes to add artificial dissipation as shown in Section 2.3. This choice can be also identified in the explicit Lax Wendroff step of Lerat Sides schemes[10] but then the flux evaluation takes into account values of w over a larger number of cells (six in two dimensions on a structured quadrilateral grid).

The second possibility is related to upwind schemes of either first or higher order. The most typical example of such upwind schemes is the basic first order Godunov scheme[18] which rests on the assumption of a piecewise constant function for representing w and on the solution of a Riemann problem at the mid-point of each interface between two cells. This Riemann problem solution provides an intermediate physical state for the flux evaluation.

If we compare the cell centered approach with the cell vertex one, it is easy to see that, in the case of quadrilateral or hexahedral grids, there is about the same number of control cells (thus of unknowns) and of flux evaluations whereas in the case of triangular or tetrahedral grids the number of control volumes (thus of unknowns) is larger in the cell centered approach with a ratio of about two for triangles and of about five or six for tetrahedra.

This is one reason why the following third class of finite volume discretization has been increasingly adopted by practitioners of finite volume methods on unstructured grids.

Node centered methods

In the node centered approach for finite volume discretization, the unknowns are associated with the mesh vertices and a control cell is constructed around each mesh vertex without overlapping neighbouring cells in a manner that provides a complete partition of the computational domain.

A strong motivation for this choice lies in the fact that for solving the Navier-Stokes equations a continuous approximation with unknowns defined at the mesh nodes is best suited for central differencing of second order derivatives. The natural practice of extending Euler flow solvers to the solution of Navier-Stokes equations is also an explanation for the interest related to this formulation which combines the choice of the nodes from cell vertex with the choice of non overlapping cells as in the cell centered approach.

The construction of the control volumes is based on the definition of a centroid in each mesh cell Then a "dual mesh" is built which connects these centroids. There are two possibilities: either two adjacent centroids are joined by a straight line ("centroid dual mesh") or a mid-point is firstly introduced on the common interface and is used (with mid-points on the edges in three dimensions) to complete the boundary of the control volume by median lines (or planes) providing the so-called "median dual mesh" (see Fig. 2.1.4).

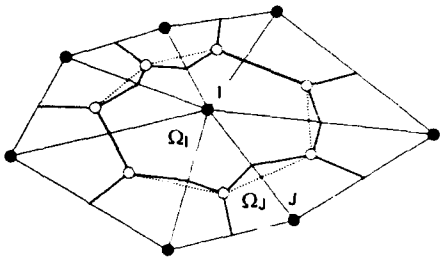


Figure 2.1.4 Node centered finite volume method:

a) ___ Median dual mesh; b) ....Centroid dual mesh.

The first possibility is more economical as flux evaluation is concerned but it can lead to large inconsistency in these flux evaluations for very distorted and elongated meshes and the second choice is recommended in that situation.

This choice of a node centered finite volume approach can be used with both central or upwind schemes but it is specially well adapted to Godunov-type methods and to unstructured grids made of triangles or tetrahedra. In this case, strong analogy can be shown[19] between a node centered finite volume method on a median dual mesh and the cell vertex or the Galerkin finite element method with linear approximation proposed by Jameson et al[20].

2.1.3 Finite Element Techniques

As indicated above, finite element methods are characterized by the use of a weak formulation as a basis for the discretization of the equations to solve.

The subdivision of the computational domain is generally

achieved with simplicial elements (triangles in 2D, tetrahedra in 3D) which made possible a linear interpolation over each element. However the use of quadrilaterals or hexahedra is also met either in combination with simplicial elements or not. The only restriction on the grid arrangement is that a vertex, edge, or face of an element has to be also a vertex, edge, or face of any adjacent element.

Finite element methods found their origin in the field of solid mechanics at the beginning of the sixties and their use for fluid dynamics appeared in the mid seventies. This extension was rather natural since at that time the first attempts to apply finite element techniques to a fluid flow problem concerned elliptic partial differential equations namely the Stokes problem and the full potential equation.

Variational principles are well known for the derivation of these equations. This is not the case for the system of the Euler equations of hyperbolic type, and it was early recognized that the direct use of the classical Galerkin approach in space-time is inefficient for these equations.

Therefore the few finite element methods existing for solving the Euler system of equations have been mostly devised from a semi- discrete approach with only a space discretization by finite element techniques and often with a strong link to well known finite difference or finite volume schemes. The noticeable exception is the Galerkin Least Square or Streamline Upwind Petrov Galerkin (SUPG) methods which have progressively evolved from research studies on scalar advection-diffusion problems to full compressible Navier-Stokes solvers. This approach is now presented by Johnson (under the name of Streamline Diffusion) as a general unified approach to CFD[21].

The presentation will be limited here to a simple overview of the main methods namely the Galerkin method with artificial dissipation of Jameson et al.[20,22], the Taylor-Galerkin method studied by Peraire, Morgan, Lohner et al.[23,24,25] after Donea[26], the Richtmyer-Galerkin approach from Angrand, Dervieux et al.[27,28] and the Galerkin Least Square after Hugues and Johnson[21].

All of these methods are based on a continuous piecewise linear approximation on a triangular or tetrahedral mesh eventhough it could be possible in theory to use higher order polynomials.

Standard Galerkin methods

We consider first the Jameson finite element method and for simplification we extract from the set of Euler equations one of the scalar conservation laws:

$$\frac{\partial w}{\partial t} + \nabla \cdot \bar{F} = 0 \qquad (2.1.22)$$

This scalar equation is then transformed with multiplication by a test function ϕ, space integration over the flow domain Ω and integration by parts for the space derivatives giving:

$$\int_{\Omega} \frac{\partial w}{\partial t}\, \phi \, d\Omega - \int_{\Omega} \bar{F}.\bar{\nabla}\phi \, d\Omega + \int_{\partial\Omega} \bar{F}.\bar{n}\, \phi \, d\sigma = 0 \quad (2.1.23)$$

We assume now that $\frac{\partial w}{\partial t}$ and $\bar{F}$ are linearly interpolated on each element from their value at nodes and that ϕ is the piecewise linear function with value unity at node I and zero at all other nodes in Ω. Then the last integral vanishes except in the case when node I is on the boundary of Ω. Also $\bar{\nabla}\phi$ is constant in every element and differs from zero only in elements sharing the common vertex I. In such an element Ω_k, shown in Fig. 2.1.5, it is easy to show that:

$$\bar{\nabla}\phi = -\frac{1}{3}\frac{S_k \bar{n}_k}{V_k}$$

where S_k and $\bar{n}_k$ are the area and the outer normal of the face opposite to node I and V_k is the volume of the element.

Thus the second integral evaluated on this element Ω_k is :

$$\int_{\Omega_k} \bar{F}.\bar{\nabla}\phi \, d\Omega = -\frac{1}{3}\frac{(\bar{F}_I + \bar{F}_J + \bar{F}_K + \bar{F}_L)}{4} \cdot \frac{S_k \bar{n}_k}{V_k} \, V_k \quad (2.1.24)$$

and since, by summation over all elements with a common vertex I, we have $\sum_k S_k \bar{n}_k = \bar{0}$,

we can discard the $\bar{F}_I$ term in the sum over all elements surrounding I getting from Eq. (2.1.24):

$$-\sum_k \int_{\Omega_k} \bar{F}.\bar{\nabla}\phi \, d\Omega = \sum_k \frac{S_k}{4}\left(\frac{\bar{F}_J + \bar{F}_K + \bar{F}_L}{3}\right) . \bar{n}_k$$

which clearly appears as a discrete approximation of :

$$\frac{1}{4}\sum_k \int_{\partial\Omega_k} \bar{F}.\bar{n} \, d\sigma$$

Finally, Eq. (2.1.23) gives for a node I:

$$\sum_k V_k \left(\frac{2}{5}\frac{dw_I}{dt} + \frac{1}{5}\sum_{M=1}^{3}\frac{dw_{kM}}{dt}\right) + \sum_k \bar{\bar{F}}_k.\bar{n}_k \, S_k = 0 \quad (2.1.25)$$

with $\bar{\bar{F}}_k = \frac{1}{3}\sum_{M=1}^{3} \bar{F}_{kM}$,

where the subscript M denotes the three exterior nodes of the k-th tetrahedron in the set of elements containing node I.

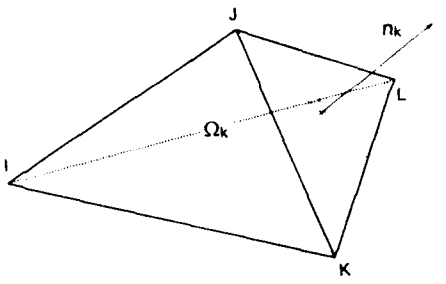


Figure 2.1.5 Tetrahedral element.

If only steady solution is of interest it is convenient to replace the "Galerkin mass matrix" (the coefficients of which appear in Eq (2.1.25)) by a "lumped mass matrix" thus avoiding the coupling of equations:

$$\left(\sum_k V_k\right)\frac{dw_I}{dt} + \sum_k \bar{\bar{F}}_k.\bar{n}_k \, S_k = 0$$

This formulation could as well have been found by a finite volume approach and this explains why it is sometimes difficult to make a distinction between the two class of discretization techniques.

Moreover, time discretizations other than the Euler explicit one are often advocated by finite element practitioners. In particular, it is worth to mention at least Richtmyer-Galerkin schemes [27] and two-step Taylor-Galerkin schemes [23,24].

These schemes are the finite element counterpart of classical predictor-corrector explicit schemes of Lax-Wendroff type in finite difference or finite volume approaches such as described by Lerat, Sides[29] or by Ni[15] .

The predictor step can be understood as a first order explicit step of cell-centered finite volume type applied on each element with a time step $\alpha\Delta t$.

$$\int_T w_k^{n+\alpha}(T)\, d\Omega = \int_T w_k^{n}\, d\Omega - \alpha\Delta t \int_{\partial T} \bar{F}_k(w^n).\bar{n}\, d\sigma$$

with $w_k^{n+\alpha}(T)$ a piecewise constant approximation.

The corrector step corresponds to the previous Galerkin finite element approximation with a time step Δt and a blending of fluxes taken at time t^n and at the end of the predictor step with a weighting factor $\frac{1}{2\alpha}$ in order to get second order accuracy.

$$\int_\Omega \frac{w^{n+1} - w^n}{\Delta t}\, \phi\, d\Omega = \int_\Omega \bar{F}(w^n).\bar{\nabla}\phi\, d\Omega + \frac{1}{2\alpha} \overset{*}{\int_\Omega} (\bar{F}(w^{n+\alpha}) - \bar{F}(w^n)).\bar{\nabla}\phi\, d\Omega$$

where $\overset{*}{\int_\Omega}$ means that a one point numerical quadrature is made on each element.

The case $\alpha = 1$ corresponds to Lax-Wendroff type schemes called Taylor-Galerkin schemes in the finite element litterature after Donea[26] and coworkers.

All these schemes give (on uniformly spaced grids) central space discretization and they need the addition of artificial viscosity in order to avoid spurious oscillations due to the decoupling of nodes or to the presence of strong gradients.

Several possibilities have been studied to give a finite element equivalent of an explicitly added artificial viscosity. Some are derived from the discretization of a Laplacian operator, other are related to upwind schemes with Roe splitting (see section 2.3) but at least one is typical of the finite element approximation. It is based on the observation that the difference between the exact Galerkin mass matrix and the lumped mass matrix is a diffusion operator which can be used in combination with a pressure switch in order to play the role of a second order dissipation near the shocks. Details can be found in Ref. 30 for a precise definition of this artificial viscosity and more generally in Ref. 25 for the class of finite element schemes of central differencing type with numerous references.

Petrov-Galerkin, Galerkin-Least-Square methods

A class of methods more specific of the finite element approach found its origin in the numerical approximation of convection-diffusion flows by Hughes and Brooks[31] with the Streamline Upwind Petrov-Galerkin (SUPG) schemes which progressively evolved towards a large family of schemes[32,33] with its theoretical analysis made by Johnson and Szepessy[34].

The space-time approximation has been developed but is mainly of interest for moving boundaries so we restrict here the presentation to the semi-discrete form of these finite element methods.

To give an idea of the different variants in this large class of discretization techniques we can restrict the problem to the case of a scalar steady advection diffusion equation:

$$Lw = \bar{a}.\bar{\nabla}w - \nabla.(k\,\bar{\nabla}w) = f$$

The standard Galerkin method consists of choosing the same approximation space V^h (generally a piecewise linear approximation) for the approximate solution w_h and for the test function v_h:

Find $w^h \in V^h$ such that for all $v^h \in V^h$

$$b(v^h, w^h) = l(v^h) \qquad (2.1.26)$$

where by definition:

$$b(v^h, w^h) = \int_\Omega (-\bar{a}.\bar{\nabla}v^h\; w^h + \bar{\nabla}v^h.\kappa\bar{\nabla}w^h)\, d\Omega$$

$$l(v^h) = \int_\Omega v^h f\, d\Omega$$

Notice that here and in the following the boundary integrals resulting from integration by parts are discarded.

The classical artificial viscosity approach, as described above, leads to replace Eq (2.1.31) by :

$$b(v^h, w^h) + \varepsilon \int_\Omega \bar{\nabla}v^h \bar{\nabla}w^h\, d\Omega = l(v^h)$$

Another possibility is to choose different approximation spaces for the numerical solution and for the test function in the weak formulation. This approach is named the Petrov-Galerkin method. An example of such a technique is the SUPG concept which relies upon:

$$b(v^h, w^h) + \int_\Omega \bar{a}.\bar{\nabla}v^h\, \tau(Lw^h - f)\, d\Omega = l(v^h)$$

A related technique is the Galerkin Least Square method which corresponds to:

$$b(v^h, w^h) + \int_\Omega Lv^h\, \tau(Lw^h - f)\, d\Omega = l(v^h)$$

The choice of the parameters ε and τ is made from theoretical analysis in order to enforce stability and accuracy.

It appears that for a purely convective system, SUPG and Galerkin Least Square are identical.

For the system of Euler equations, the necessary extension of the above formulation led to the introduction of rather complicated non linear operators[33,36].

Another originality in the formulation of those methods is the use of entropy variables. Indeed, starting from the observation that the $L^2(\Omega)$ inner product for trying to derive a variational statement directly from the Euler equations has no physical meaning:

$$\int_\Omega \mathbf{w}'\left(\frac{\partial \mathbf{w}}{\partial t} + \cdots\right) d\Omega = \frac{1}{2}\frac{\partial}{\partial t}\int_\Omega \rho^2(1 + u^2 + e^2)\, d\Omega + \ldots$$

the authors[32] advocate the use of entropy variables allowing a symmetrization of the system of Euler equations[35]. They give a formulation comprising the full compressible Navier-Stokes equations which accounts for a continuous approximation and a non divergence form of the convective terms and which can be written (here with repeated indices indicating summation):

$$\frac{\partial \mathbf{w}}{\partial t} + \mathbf{A}_i \frac{\partial \mathbf{w}}{\partial x_i} = \frac{\partial}{\partial x_i}\left(\mathbf{K}_{ij}\frac{\partial \mathbf{w}}{\partial x_j}\right) + \mathbf{f} \qquad (2.1.27)$$

where $\mathbf{K}_{ij} = \mathbf{K}_{ij}{}^{v} + \mathbf{K}_{ij}{}^{h}$,

and $\mathbf{K}_{ij}{}^{v}$ corresponds to the physical viscosity and $\mathbf{K}_{ij}{}^{h}$ to the heat coefficients.

The entropy variables are defined by:

$$\mathbf{u}' = \frac{dS}{d\mathbf{w}} \quad \text{where} \quad S = -\rho\, s = -\rho\, Log\left(\frac{p}{p_0}\left(\frac{\rho}{\rho_0}\right)^{-\gamma}\right)$$

The change of variables $\mathbf{w} = \mathbf{w}(\mathbf{u})$ gives:

$$\mathbf{A}_0 \frac{\partial \mathbf{u}}{\partial t} + \tilde{\mathbf{A}}_i \frac{\partial \mathbf{u}}{\partial x_i} = \frac{\partial}{\partial x_i}(\tilde{\mathbf{K}}_{ij} \frac{\partial \mathbf{u}}{\partial x_j}) + \mathbf{f} \qquad (2.1.28)$$

where $\mathbf{A}_0 = \frac{\partial \mathbf{w}}{\partial \mathbf{u}}$; $\tilde{\mathbf{A}}_i = \mathbf{A}_i \mathbf{A}_0$; $\tilde{\mathbf{K}}_{ij} = \mathbf{K}_{ij} \mathbf{A}_0$.

The coefficient matrices possess the following properties:

$\mathbf{A}_0$ is symmetric positive definite, the $\mathbf{A}_i$ are symmetric and the $\mathbf{K}_{ij}$ are symmetric positive semi-definite.

A dot product of Eq (2.1.28) by $\mathbf{u}$ gives after integration:

$$0 = \int_\Omega \mathbf{u}^t \, (\mathbf{A}_0 \frac{\partial \mathbf{u}}{\partial t} + \tilde{\mathbf{A}}_i \frac{\partial \mathbf{u}}{\partial x_i} - \frac{\partial}{\partial x_i}(\tilde{\mathbf{K}}_{ij} \frac{\partial \mathbf{u}}{\partial x_j}) - \mathbf{f}) \, d\Omega$$

$$= \int_\Omega (\frac{\partial S}{\partial t} + \frac{\partial}{\partial x_i}(u_i S) + (\frac{\partial \mathbf{u}}{\partial x_i})^t \tilde{\mathbf{K}}_{ij} \frac{\partial \mathbf{u}}{\partial x_j}$$

$$- \frac{\partial}{\partial x_i}(\mathbf{u}^t \tilde{\mathbf{K}}_{ij} \frac{\partial \mathbf{u}}{\partial x_j}) - \mathbf{u}^t \mathbf{f}) \, d\Omega \qquad (2.1.29)$$

that leads to a Clausius-Duhem type inequality or second law of thermodynamics[12].

$$\int_\Omega (\frac{\partial \rho s}{\partial t} + \frac{\partial}{\partial x_i}(\rho u_i s) + \frac{\partial}{\partial x_i}(\frac{q_i}{c_v T})) \, d\Omega =$$

$$\int_\Omega \frac{\partial \mathbf{u}^t}{\partial x_i} \tilde{\mathbf{K}}_{ij} \frac{\partial \mathbf{u}}{\partial x_j} \, d\Omega \geq 0$$

where $\bar{q}$ is the heat flux vector.

For the Euler equations Eq.(2.1.29) gives:

$$\int_\Omega \mathbf{u}^t \, (\mathbf{A}_0 \frac{\partial \mathbf{u}}{\partial t} + \tilde{\mathbf{A}}_i \frac{\partial \mathbf{u}}{\partial x_i}) \, d\Omega =$$

$$\int_\Omega (\frac{\partial \rho s}{\partial t} + \frac{\partial}{\partial x_i}(\rho u_i s)) \, d\Omega = 0$$

showing that a Galerkin formulation based on the entropy variables with a continuous approximation cannot lead to an entropy production.

This is a justification for resorting to a Petrov-Galerkin formulation:

Find $\mathbf{u}$ such that for any $\bar{\mathbf{v}}$

$$\int_\Omega \bar{\mathbf{v}}^t \, (\mathbf{A}_0 \frac{\partial \mathbf{u}}{\partial t} + \tilde{\mathbf{A}}_i \frac{\partial \mathbf{u}}{\partial x_i}) \, d\Omega = 0 \qquad (2.1.30)$$

with $\bar{\mathbf{v}} = \mathbf{v} + \mathbf{T}_i \frac{\partial \mathbf{v}}{\partial x_i}$; $\mathbf{T}_i = \mathbf{T}_i^{(1)} + \mathbf{T}_i^{(2)}$,

where $\mathbf{T}_i^{(1)}$ is a streamline diffusion matrix and $\mathbf{T}_i^{(2)}$ a discontinuity capturing operator. A precise description of these matrices is beyond the scope of this short presentation and more details may be found in the references mentionned above. Implementation of such Galerkin Least Square finite element methods has been made for complex high speed flows[37].

To conclude this subsection on finite element methods it is worth to say a few words about the practical use of these techniques. Firstly, we insist on the fact that for Euler solvers it is not legitimate to identify every unstructured grid method with a finite element one since most of the unstructured grid methods are of finite volume type[38,39,40] specially for upwind schemes. The main advantage of unstructured grids is based on their ability to represent complex geometries however with a need of some effort for a priori controlling the distribution of nodes. A strong increase of interest resulted from the introduction of adaptive mesh techniques by local refinement and coarsening of the grid according to some flow solution error indicator. This can favourably counterbalance the extra cost of unstructured grid solution algorithms and programming techniques.

The effectiveness of the adaptive grid methodology depends on the quality and accuracy which can be reached for "a posteriori error indicators" and it is in that direction that finite element analysis could bring powerful promising techniques.

2.1.4 References

1. WOODWARD, P.R. and COLELLA, P. "The Piecewise Parabolic Method for Gas Dynamics Calculations". Journal of Computational Methods, 54, pp. 174-201, 1981.

2. HARTEN, A. and OSHER, S. "Uniformly High Order Accurate Non-Oscillatory Schemes, I.". MRC Technical Summary Report 2823, 1985.

3. HARTEN, A., ENGQUIST, B., OSHER, S. and CHAKRAVARTHY, S.R. "Uniformly High Order Accurate Essentially Non-Oscillatory Schemes, III". ICASE Report 86-22, 1986.

4. DECONINCK, H., STRUIJS, R. and ROE, P.L. "Fluctuation Splitting for Multi-Dimensional Convection Problems : an Alternative to Finite Volume and Finite Element Methods". *Von Karman Institute Lecture Series 1990-03*, 1990.

5. BEAM, R.M. and WARMING, R.F. "Implicit Numerical Methods for the Compressible Navier-Stokes and Euler Equations". *Von Karman Institute Lecture Series 1982-04*, 1982.

6. JAMESON, A., SCHMIDT, W. and TURKEL, E. "Numerical Simulation of the Euler Equations by Finite Volume Methods Using Runge-Kutta Time Stepping Schemes". AIAA Paper 81- 1259, AIAA 5th CFD Conference, 1981.

7. BRILEY, W.R. and McDONALD, H. "Solution of the Three-Dimensional Navier-Stokes Equations by an Implicit Technique". *Proc. of the Fourth International Conference on Numerical Methods in Fluid Dynamics, Lecture Notes in Physics, Vol. 35* , Berlin, Springer, 1975.

8. LERAT, A. "Difference Methods for Hyperbolic Problems with Emphasis on Space Centered Approximation ". *Von Karman Institute Lecture Series 1990-03* , 1990.

9. LERAT, A. " Implicit Methods of Second Order Accuracy for the Euler Equations". AIAA Paper 83-1925 and AIAA Journal 23, pp. 33-40, 1985.

10. LERAT, A. and SIDES, J. "Efficient Solution of the Steady Euler Equations with a Centered Implicit Method". in " *Numerical Methods in Fluid Dynamics III*", K.W. Morton and M.J. Baines Eds., Clarendon Press, Oxford, pp. 65-86, 1988.

11. VIVIAND, H. "Conservative Forms of Gas Dynamics Equations", La Recherche Aerospatiale, 1974-1, p. 65, 1974.

12. VINOKUR, M. "Conservation Equations of Gas Dynamics in Curvilinear Coordinate Systems". Journal of Computational Physics, 14, pp. 105-125, 1975.

13. McDONALD, P.W. "The Computation of Transonic Flow through Two Dimensional Gas Turbine Cascades". ASME Paper 71-GT-89, 1971.

14. MacCORMACK, R.W. and PAULLAY, A.J. "Computational Efficiency Achieved by Time Splitting of Finite Difference Operators". AIAA Paper 72-154, San Diego, 1972.

15. NI, R.N. "A Multigrid Scheme for Solving the Euler Equations". AIAA Journal, 20-11, pp. 1565-1571, 1982.

16. HALL, M.G. "Cell Vertex Multigrid Scheme for the Solution of the Euler Equations". *Proc. of the Conference on Numerical Methods in Fluid Dynamics*, Reading U.K., 1985.

17. RIZZI, A., ERIKSSON, L.E. "Computation of Flow around Wings Based on the Euler Equations". Journal of Fluid Mechanics, Vol. 148, pp. 45-71, 1984.

18. GODUNOV, S.K., ZABRODINE, A., IVANOV, M., KRAIKO, A., PROKOPOV, G. "*Resolution numerique des problemes multidimensionnels de la dynamique des gaz*". Moscow, Editions MIR.

19. BARTH, T.J. "On Unstructured Grids and Solvers". *Von Karman Institute Lecture Series 1990-03* , 1990.

20. JAMESON, A., BAKER, T.J., WEATHERILL, N.P. "Calculation of Inviscid Transonic Flow over a Complete Aircraft". AIAA Paper 86-0103, Reno, 1986.

21. JOHNSON,C. "Finite Element Methods for Flow Problems". *AGARD Special Course on Unstructured Grid Methods for Advection Dominated Flows, AGARD-R-787* , Paper 1, May 1992.

22. JAMESON,A., BAKER, T.J. "Improvements to the Aircraft Euler Method". AIAA Paper 87-0452, AIAA 25th Aerospace Sciences Meeting, Reno, January 1987.

23. PERAIRE J., PEIRO J., FORMAGGIA L., MORGAN K., ZIENKIEWICZ O.,C. "Finite Element Euler Computations in Three Dimensions" Int. Journal for Numerical Methods in Engineering, Vol 26, pp 2135-2159, 1988.

24. LOHNER R., MORGAN K., ZIENKIEWICZ O.,C. "The Solution of Nonlinear Systems of Hyperbolic Equations by the Finite Element Method". Int. Journal for Numerical Methods in Fluids, Vol 4, pp 1043-1063, 1984.

25. MORGAN K., PERAIRE J., PEIRO J. "Unstructured Grid Methods for Compressible Flows". *AGARD Special Course on Unstructured Grid Methods for Advection Dominated Flows, AGARD-R-787* , Paper 5, May 1992.

26. DONEA J. "A Taylor-Galerkin Method for Convective Transport Problems". Int. Journal in Numerical Methods in Engineering, Vol 20, pp 101-119, 1984.

27. ANGRAND F., DERVIEUX A., LOTH L., VIJAYASUNDARAM G. "Simulation of Euler Transonic Flows by Means of Explicit Finite Element Type Schemes" INRIA Research Report 250, 1983.

28. ARMINJON P., DERVIEUX A., FEZOUI L., STEVE H., STOUFFLET B. in *Notes on Numerical Fluid Mechanics 24* , Eds R.Jeltsch and J. Ballmann, Vieweg Braunschweig, 1988.

29. LERAT A., SIDES J. "A New Finite Volume Method for the Euler Equations with Application to the Transonic Flows" in *Numerical Methods in Aeronautical Fluid Dynamics* ,P.L. Roe Ed, Academic Press, 1982.

30. MORGAN K., PERAIRE J. "Finite Element Methods for Compressible Flows" *von Karman Institute for Fluid Dynamics, Lecture Series 1987-04* , 1987.

31. HUGHES T.J.,R., BROOKS A. "A Multidimensional Upwind Scheme with no Crosswind Diffusion". *AMD-Vol 34, Finite Element Methods for Convection Dominated Flows*, Ed. Hughes T.J.,R., ASME New York, 1979.

32. HUGHES T.J.,R., FRANCA L.,P., MALLET M. "A new Finite Element Formulation for Computational Fluid Dynamics: I Symmetric forms of the Compressible Euler and Navier-Stokes Equations and the Second Law of Thermodynamics". Computer Methods in Applied Mechanics and Engineering, Vol 54, pp 223-234, 1986.

33. HUGHES T.J.,R., MALLET M., MIZUKAMI A. "A new Finite Element Formulation for Computational Fluid Dynamics: II Beyond SUPG". Computer Methods in Applied Mechanics and Engineering, Vol 54, pp 341-355, 1986.

34. JOHNSON C., SZEPESSY A. "On the Convergence of Streamline Diffusion Finite Element Methods for Hyperbolic Conservative Laws" in *Numerical Methods for Compressible Flows - Finite Difference, Element and Volume Techniques, T.E. Tezduyar and T.J.R. Hughes (Eds), AMD Vol 78*, pp 75-91, ASME, New York, 1986.

35. HARTEN A. "On the Symmetric Form of Systems of Conservation Laws with Entropy". Journal of Computational Physics, Vol 49, pp 151-164, 1983.

36. SHAKIB F. "*Finite Element Analysis of the Compressible Euler and Navier-Stokes Equations*". Ph D. Thesis, Stanford University, 1988.

37. BRISTEAU M., O., MALLET M., PERIAUX J., ROGE G. "Development of Finite Element Methods for Compressible Navier-Stokes Flow Simulations in Aerospace Design", AIAA Paper 90-0403, 28th AIAA Aerospace Sciences Meeting, Reno, Jan. 1990.

38. BARTH T.,J. "Aspects of Unstructured Grids and Finite Volume Solvers for the Euler and Navier-Stokes Equations". *AGARD Special Course on Unstructured Grid Methods for Advection Dominated Flows, AGARD-R-787* , Paper 6, May 1992.

39. VANKEIRSBILCK P., DECONINCK H. "Higher Order Upwind Finite Volumes Schemes with ENO Properties for General Unstructured Meshes". *AGARD Special Course on Unstructured Grid Methods for Advection Dominated Flows, AGARD-R-787* , Paper 7, May 1992.

40. FEZOUI L., STOUFFLET B. "A Class of Implicit Upwind Schemes for Euler Simulation with Unstructured Meshes". Journal of Computational Physics, Vol 84-1, pp 174-206, Sept. 1989.

2.2 COMPUTATIONAL GRIDS

In Chapter One the conservation laws relating to mass, momenta and energy were described in mathematical form. These conservation statements assume the continuum hypothesis, in which a gas is assumed to be a continuous medium, rather than made up from individual atoms and molecules. The mathematical equations are presented in both integral and differential form, but both forms assume that the variables are defined continuously through the flow domain. The conservation equations, for the majority of flows of interest, are non-linear and hence are not amenable to classical mathematical analysis. Alternative methods of solution have to be sought and one of these approaches is a numerical solution of the equations. In the previous section of this chapter, several of the major techniques for representing equations, in a form required for numerical analysis, were presented. All of these methods assume that the flow domain has been spatially discretized into a grid or mesh which consists of a set of points, and connections between points, called cells or elements. The process of spatial discretization is called grid or mesh generation.

The spatial discretization plays an important role in the numerical analysis procedure. The distribution of points in the domain must, firstly, adequately represent the geometry of the region. In general, regions of the boundary of the domain which change rapidly will require a concentration of points to adequately represent the shape. Secondly, for accurate flow simulations, it is necessary that there are adequate points in regions of the flowfield where there is high activity, i.e., where the flow variables are changing rapidly. These two major requirements cannot be achieved in an arbitrary way. The accuracy of the numerical discretizations are dependent upon the properties of the underlying spatial discretization. For example, the classical second order accurate finite difference representation of a second derivative is only second order in space if the grid point spacing is uniform. Furthermore, it can be shown that other spatial derivatives only achieve second order accuracy if the grid lines are approximately orthogonal. Hence, the spatial discretization of the domain must be achieved without discontinuous grid point spacing and without the introduction of highly skewed cells or elements. These restrictions make the generation of suitable computational grids a non-trivial problem. Considerable attention in the last decade has been given to the problem of grid generation[1-7].

Before discussing, in detail, some of the more popular and successful approaches, it is necessary to comment on different methodologies used in grid generation. One of the first aspects to cover is the spatial pattern and arrangement of points and cells.

Structured and Unstructured Grids

The basic difference between structured and unstructured grids lies in the form of the data structure which most appropriately describes the grid. A structured grid of quadrilaterals consists of a set of coordinates and connectivities which naturally map into elements of a matrix. Neighbouring points in a mesh in the physical space are the neighbouring elements in the mesh point matrix (Fig. 2.2.1).

Thus, for example, a 2-dimensional array x(i,j) can be used to store the x-coordinates of points in a 2D grid. The index i can be chosen to describe the position of points in one direction, whilst j describes the position of points in the other direction. Hence, in this way, the indices i and j represent the 2 families of curvilinear lines. These ideas naturally extend to 3 dimensions.

However, for an unstructured mesh the points cannot be represented in such a manner and additional information has to be provided. For any particular point, the connection with other points must be defined explicitly in the connectivity matrix. (Figure 2.2.2)

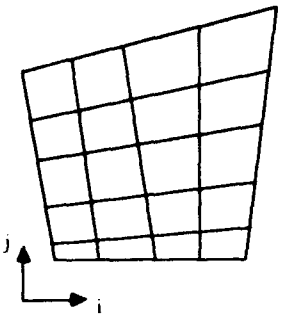
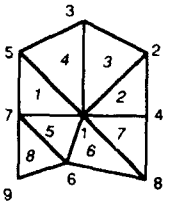


Figure 2.2.1 Data ordering of a structured grid. Quadrilaterals formed by $(r_{ij}, r_{i+1j}, r_{i+1j+1}, r_{ij+1})$ (i=1,4, j=1,5).



Element	Nodes		
1	1.	5.	7
2	1.	4.	2
3	1.	2.	3
4	1.	3.	5
5	1.	7.	6
6	1.	6.	8
7	1.	8.	4
8	7.	9.	6

Figure 2.2.2 Data ordering of an unstructured grid

A typical form of data format for an unstructured grid in 2 dimensions is;

Number of points, number of elements
x_1, y_1
x_2, y_2
x_3, y_3
.....

n_1, n_2, n_3
n_4, n_5, n_6
n_7, n_8, n_9
.....

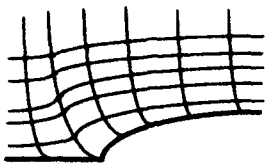
where (x_i, y_i) are the coordinates of point i, and n_i, 1=1,N are the point numbers with, for example, the triad (n_1, n_2, n_3) forming a triangle. Other forms of connectivity matrices are equally valid, for example, connections can be based upon edges.

The real advantage of the unstructured mesh is, however, because the points and connectivities do not possess any global structure. It is possible, therefore, to add and delete nodes and elements as the geometry requires or, in a flow adaptivity scheme, as flow gradients or errors evolve. Hence the unstructured approach is ideally suited for the discretization of complicated geometrical domains and complex flowfield features. However, the lack of any global directional features in an unstructured grid makes the application of line sweep solution algorithms more difficult to apply than on structured grids.

Boundary Conforming and Non-Aligned Grids

The solid boundaries within a flowfield play a vitally important role. In most applications the solid boundaries create the flow features of interest. Hence, it is essential that the solid boundaries are accurately represented in the spatial grid and then the numerical formulation of the solid wall boundary conditions can be implemented accurately and efficiently. These

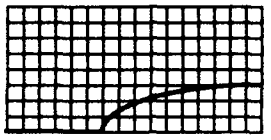
requirements have resulted in the use of boundary or body conforming grids, in which points lie on the boundaries and grid lines are aligned with the geometrical surfaces (Figure 2.2.3).



Solid Boundary

Figure 2.2.3 Boundary conforming grid

Such an approach, although probably the natural approach from the viewpoint of the implementation of the flow boundary conditions, places a severe restriction on grid generation procedures. An alternative approach[8-11], is to relax this restriction on the grid generation and allow grid lines to pass through the solid boundary in a non-aligned manner (Fig 2.2.4).



Solid Boundary

Figure 2.2.4 Non-aligned grid

The problem of the accurate imposition of boundary conditions is then transferred to the solution algorithm and the construction of appropriate techniques.

Both the non-aligned and boundary conforming approaches have been investigated. Today the boundary conforming strategy is the most popular and most widely used.

One of the features of constructing grid generation techniques, which is different from the development of flow or other analysis algorithms, is that there are no governing laws which govern grid generation. Any suitable empirical or geometrical constructions can be used. This is reflected in the many different and diverse techniques which have been explored. However, after a period of exploration and numerical experimentation, several approaches are now becoming standard procedures. The major techniques will be discussed.

2.2.1 *Structured Grids from* Partial Differential Equations

Elliptic Systems

The motivation for the use of elliptic equations as generators of grid points can be derived from a number of sources. The nature of elliptic equations is to smooth boundary data and this affords a most desirable property. Laplace's equation with Cauchy-Riemann type boundary conditions can be used to generate conformal mappings. In fact, the real and imaginary parts of an analytic transformation are harmonic functions. An alternative viewpoint, and one which is most appropriate in computational fluid dynamics, is to note that inviscid steady incompressible flow is described in terms of Laplace's equation in the potential function and the stream function.

A starting point for elliptic equations is to choose a system of Poisson equations, expressed in the form

$$\begin{aligned} \xi_{xx} + \xi_{yy} + \xi_{zz} &= P(\xi,\eta,\zeta) \\ \eta_{xx} + \eta_{yy} + \eta_{zz} &= Q(\xi,\eta,\zeta) \\ \zeta_{xx} + \zeta_{yy} + \zeta_{zz} &= R(\xi,\eta,\zeta) \end{aligned} \qquad (2.2.1)$$

in which P, Q and R are the source functions which can be used to control the grid point spacing and distribution. These equations, as expressed in Eq. (2.2.1), are in an inappropriate form for grid generation, since in general, boundary data will be specified in terms of x, y and z and not ξ, η, and ζ. Hence, following Thompson, Thames and Mastin[12], these equations are transformed so that they are written in terms of the unknown spatial physical variables x,y and z with the independent variables ξ, η, and ζ. The transformation leads to the equations

$$\begin{aligned} \alpha_{11} r_{\xi\xi} + \alpha_{22} r_{\eta\eta} + \alpha_{33} r_{\zeta\zeta} + 2(\alpha_{12} r_{\xi\eta} + \alpha_{13} r_{\xi\zeta} + \alpha_{23} r_{\eta\xi}) \\ = -J^2 (P r_{\xi} + Q r_{\eta} + R r_{\zeta}) \end{aligned} \qquad (2.2.2)$$

where $r = (x,y,z)^T$, $\alpha_{ij} = \sum \gamma_{mi} \gamma_{mj}$ and γ_{ij} is the ij-th cofactor of the matrix

$$M = \begin{vmatrix} x_\xi & x_\eta & x_\zeta \\ y_\xi & y_\eta & y_\zeta \\ z_\xi & z_\eta & z_\zeta \end{vmatrix} \qquad (2.2.3)$$

and the Jacobian J is the determinant of M.

An interesting alternative derivation of these equations was given by Brackbill and Saltzman[13]. Working on adaptive grids, they introduced global smoothness as one property required from the mesh. Smoothness, in a 2-dimensional domain Ω with boundary Σ, can be written as

$$I_S = \int_{\Sigma} |(\nabla\xi)^2 + (\nabla\eta)^2| \, d\Omega,$$

which when transformed and optimized leads to the 2-dimensional equations equivalent to those expressed in Eq. (2.2.2). Hence, the inverted Laplacian, without grid control functions, maximises the smoothness of the distribution of grid points.

The above equations represent a non-linear boundary value problem. The solution of these equations can be achieved using any appropriate technique. However, central differences and relaxation schemes are commonly used. To illustrate a solution procedure, consider the commonly used 2-dimensional form, $r = r(x,y)^T$, of the equations with the control functions set to zero, namely,

$$\alpha r_{\xi\xi} - 2\beta r_{\xi\eta} + \gamma r_{\eta\eta} = 0,$$

where

$$\alpha = (x_\eta^2 + y_\eta^2),$$

$$\beta = (x_\xi x_\eta + y_\xi y_\eta)$$

and

$$\gamma = (x_\xi^2 + y_\xi^2)$$

Using central finite differences for the representation of the derivatives, the residual on a square mesh with $\xi = ih$, $(i=0,1,2,.....m)$ and $\eta = jh$, $(j=0,1,2,.....n)$ can be represented as

$$R_{ij}^n = \alpha_{ij}^n (r_{i+1j}^n - 2r_{ij}^n + r_{i-1j}^n) - 2\beta_{ij}^n(r_{i+1j+1}^n - r_{i-1j+1}^n - r_{i+1j-1}^n + r_{i-1j-1}^n)/4 + \gamma_{ij}^n(r_{ij+1}^n - 2r_{ij}^n + r_{ij-1}^n)$$

with

$$\alpha_{ij}^n = [(x_{ij+1}^n - x_{ij-1}^n)^2 + (y_{ij+1}^n - y_{ij-1}^n)^2]/4$$

$$\beta_{ij}^n = [(x_{i+1j}^n - x_{i-1j}^n)(x_{ij+1}^n - x_{ij-1}^n) + (y_{i+1j}^n - y_{i-1j}^n)(y_{ij+1}^n - y_{ij-1}^n)]/4$$

$$\gamma_{ij}^n = [(x_{i+1j}^n - x_{i-1j}^n)^2 + (y_{i+1j}^n - y_{i-1j}^n)^2]/4$$

where r_{ij}^n represents the unknown at the point (i,j) at iteration level n. It is noted that in the numerical formulation a linearization is applied to the terms in α, β, and γ. In practice, once x and y are known, at the new iteration level n+1, they can be immediately utilised in the terms α_{ij}^n, β_{ij}^n and γ_{ij}^n. It follows that the solution to the equations can be obtained using the stationary and linear successive over-relaxation scheme.

$$r_{ij}^{n+1} = r_{ij}^n + \omega \frac{R_{ij}^n}{2(\alpha_{ij}^n + \gamma_{ij}^n)}$$

where ω is a relaxation parameter, with $1<\omega<2$. A similar numerical scheme can be applied to the equations, Eq. (2.2.2).

Boundary conditions for these equations are most generally applied in the form of Dirichlet or Neumann conditions. Dirichlet conditions specify fixed (x,y) values for a particular constant ξ or η and are used, for example, to specify aerofoil or outerboundary positions. Neumann conditions are used to specify fixed derivatives on boundaries. In the case of the grid equations, this type of boundary condition amounts to the specification of grid line directions. The positions of (x,y), on a boundary of constant ξ and η, are allowed to move in order to satisfy the slope conditions. Typically, such boundary conditions are applied on lines where one of the coordinates x or y is fixed whilst the other is computed under the Neumann condition. Such a boundary condition may be applied at an outer boundary or on a cut in the wake of an aerofoil. Similar boundary conditions are applied in 3 dimensions.

Central to the practical use of these equations is the appropriate form for the control functions P, Q and R. The inherent smoothing properties of Laplace's equation ensures that, in the absence of boundary curvature, the grid points are evenly spaced. However, near convex boundaries the grid points will become more closely spaced, whilst near concave boundaries the mesh spacing will be more sparse. These properties are not always desirable for grid generation where it is essential that the grid near a boundary reflects the shape of that boundary. Control of grid point spacing can be achieved by the source terms P, Q and R.

Typical effects of the source functions, in 2 dimensions, are shown in schematic form in Fig. 2.2.5.

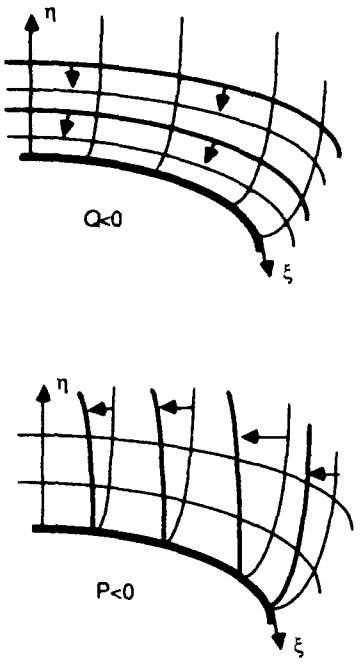


Figure 2.2.5. Effects of the control functions

Negative values of Q tend to cause the η-coordinate lines to move in the direction of decreasing η, while negative values of P cause ξ-lines to move in the direction of decreasing ξ. Similar effects occur in 3 dimensions.

Automatic procedures for the computation of the source terms have been devised[14]. A popular approach is to derive the control functions from the boundary point spacing and then to interpolate the values into the interior. In this way, point spacing in the field reflects the point spacing on boundaries[15].

To illustrate some of these ideas the equations in 2 dimensions, which include the control terms, are

$$\alpha(r_{\xi\xi}+Pr_\xi) - 2\beta r_{\xi\eta} + \gamma(r_{\eta\eta}+Qr_\eta) = 0. \qquad (2.2.4)$$

The work of Thomas and Middlecoff[15] demonstrated that the distribution of points on boundaries of a domain can be used to generate the control functions so as to have the effect of extending the point distribution into the interior. Eliminating Q in Eq. (2.2.4) leads to

$$\alpha[y_\eta(x_{\xi\xi}+Px_\xi) - x_\eta(y_{\xi\xi}+Py_\xi)] = y_\eta^2 [\, 2\beta (\frac{x_\eta}{y_\eta})_\xi + \gamma (\frac{x_\eta}{y_\eta})_\eta] \qquad (2.2.5)$$

Imposing two conditions on an η=constant boundary, namely, zero curvature,

$$(\frac{x_\eta}{y_\eta})_\eta = 0 ,$$

and the condition for orthogonality

$$x_\xi x_\eta + y_\xi y_\eta = 0 ,$$

Eq. (2.2.5) reduces to

$$P = - \frac{r_\xi \cdot r_{\xi\xi}}{|r_\xi|^2} \qquad (2.2.6a)$$

A similarly expression can be derived for Q

$$Q = - \frac{r_\eta \cdot r_{\eta\eta}}{|r_\eta|^2} \qquad (2.2.6b)$$

It is noted that the evaluation of P involves the point distribution on a line of constant ξ, whilst Q involves data on a constant η. These forms for P and Q can thus be evaluated directly from the boundary point distributions.

Once values for P and Q have been obtained for each boundary point the values at interior mesh points can be obtained by linear interpolation along lines of constant ξ and η. This procedure ensures that the grid throughout the interior of the domain reflects the distribution of points on the boundaries. This effect is illustrated in Fig. 2.2.6.

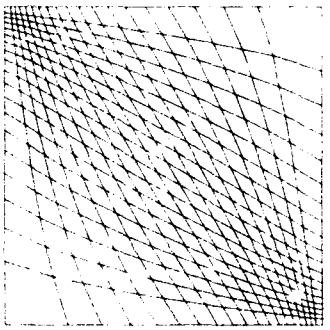


Figure 2.2.6 Effects of control functions on the spacing of points inside the domain. The interior point distribution reflects the point spacing on the boundary.

Although this approach works well in many cases there are circumstances when local control of orthogonality in the mesh is important. This is particularly relevant for meshes to be used for viscous flow simulation. To implement local orthogonality, a further modification to the computation of the control terms can be made[14].

For orthogonality, $\beta = 0$ and thus the grid equations become

$$\alpha(r_{\xi\xi}+Pr_\xi) + \gamma(r_{\eta\eta}+Qr_\eta) = 0.$$

Taking the scalar product of this equation with r_ξ and r_η and again using the condition for orthogonality leads to

$$P = - \frac{r_\xi \cdot r_{\xi\xi}}{|r_\xi|^2} - \frac{r_\xi \cdot r_{\eta\eta}}{|r_\eta|^2} \qquad (2.2.7a)$$

and

$$Q = - \frac{r_\eta \cdot r_{\eta\eta}}{|r_\eta|^2} - \frac{r_\eta \cdot r_{\xi\xi}}{|r_\xi|^2} \qquad (2.2.7b)$$

The first terms of Eq. (2.2.7) for P and Q are equivalent to the control functions of Thomas and Middlecoff. However, the second terms are the corrective terms for orthogonality which are not only dependent upon the boundary points but also on the field points. Hence, it is not possible to use these equations without some interaction with the grid as it evolves in the solution procedure.

The iterative procedure for the control functions then follows the steps:-

1. *Compute P and Q from the boundary point distribution*

$$P = - \frac{r_\xi \cdot r_{\xi\xi}}{|r_\xi|^2} \qquad Q = - \frac{r_\eta \cdot r_{\eta\eta}}{|r_\eta|^2}$$

2. Obtain an initial grid.

3. Calculate the corrective terms

$$P' = - \frac{r_\xi \cdot r_{\eta\eta}}{|r_\eta|^2} \quad \text{and} \quad Q' = - \frac{r_\eta \cdot r_{\xi\xi}}{|r_\xi|^2}$$

4. *Add to the control terms*

$$P = P + P' \text{ and } Q = Q + Q'$$

5. Solve Eq. (2.2.4) to produce a new grid

6. Repeat steps 3,4,5 until the correction terms P' and Q' are zero.

It is clear that the method adds correction terms for orthogonality until they are zero. Fig. 2.2.7 shows the effects of this iterative control of the source terms.

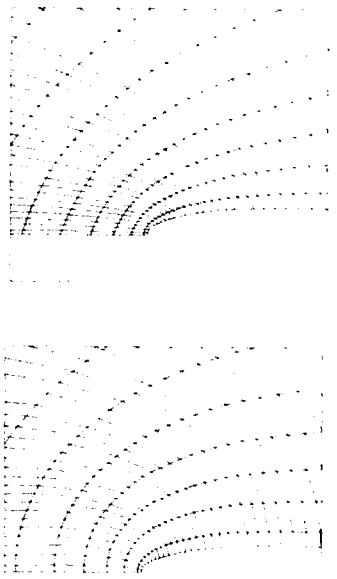


Figure 2.2.7 Effect of the boundary orthogonality control.

Further discussions of the approach and the specification of the control terms can be found elsewhere[14].

Other elliptic partial differential equations have been used for grid generation[16]. Higher order systems have been used since they provide the flexibility for specifying more conditions on the boundary. In this way, it is possible to not only specify point spacing, but also slopes etc..

Hyperbolic Systems

Computational grids can also be generated by solving hyperbolic partial differential equations which are marched outwards from a specified distribution of points on an inner boundary[17]. Differential constraints on mesh size and angles are generally used to determine the governing grid generation equations. Because the outer boundary of a grid cannot be specified in hyperbolic grid generation, the method is best suited to applications which have asymptotic-like outer boundaries. In external aerodynamics, for example, the outer boundary is generally a farfield condition of undisturbed flow. However, domain decomposition methods that use overset composite meshes for complex field simulations can also effectively employ hyperbolic grid generation. For example, a number of recent flow simulations which employ the ' Chimera' overset grid method have relied almost exclusively on hyperbolic grid generation[18,19].

In three-dimensional applications of hyperbolic grid generation, a body surface is chosen to coincide with $\zeta(x,y,z) = 0$, ζ is used as the marching direction, and the outer boundary $\zeta(x,y,z) = \zeta_{max}$ is not specified. On the body surface, grid-line distributions of ξ=constant and η=constant are user-specified. In three-dimensions, there are three orthogonality relations to choose from and one cell volume constraint. Because ζ is the marching direction, it is natural to use only the two orthogonality relations that involve ζ. This leads to the governing equations

$$r_\xi \cdot r_\zeta = x_\xi x_\zeta + y_\xi y_\zeta + z_\xi z_\zeta = 0,$$

$$r_\eta \cdot r_\zeta = x_\eta x_\zeta + y_\eta y_\zeta + z_\eta z_\zeta = 0,$$

$$\left|\frac{\partial(x,y,z)}{\partial(\xi,\eta,\zeta)}\right| = x_\xi y_\eta z_\zeta + x_\zeta y_\xi z_\eta + x_\eta y_\zeta z_\xi - x_\xi y_\zeta z_\eta - x_\eta y_\xi z_\zeta - x_\zeta y_\eta z_\xi = \Delta V$$

with r is defined as $(x,y,z)^T$. The first two equations represent orthogonality relations between ξ and ζ and between η and ζ, and the last equation is the volume or finite Jacobian constraint. ΔV can be prespecified to give appropriate point spacing.

The equations comprise a system of nonlinear differential equations in which x, y,and z are specified as initial data at ζ=0. A local linearization is performed and the equations are then solved with a noniterative approximately factored implicit finite difference scheme so that the marching step size, in ζ, can be arbitrary selected. Further details of this approach can be found in Steger[17].

Parabolic Systems

Grid generation based upon parabolic partial differential equations can also be performed[20]. In general, such equations are constructed by modifying elliptic generation systems so that the second derivatives, in one coordinate direction, do not appear. The solution of the parabolic equations can then be marched away from a boundary in much the same way as that previously described for hyperbolic grid generation. Here, however, some influence of the other boundaries is retained in the equations. Although some development of such schemes has been made, it has proved to be limited in its flexibility and applicability.

2.2.2 Structured Grids from Algebraic Methods

Algebraic grid generation distinguishes itself from other grid generation methodologies by the ability to provide a direct functional description of the coordinate transformations between the computational and physical domains. The roots of algebraic grid generation are found in conformal mapping, defined by explicit analytical functions of a complex variable.

Conformal Mapping

The concept of a mapping, in 2 dimensions, is to define a transformation which takes a domain D, defined in the plane (x,y) onto a rectangular domain R on the plane (ξ,η). The geometrical relationship between D and R is described by the components of the metric tensor g_{ij} (i,j=1,2). From the formal definition of a conformal mapping[21] it follows that the components of the metric tensor are subject to the constraint

$$g_{11} = g_{22} \text{ and } g_{12} = 0$$

and as such the mapping functions $x(\xi,\eta)$ and $y(\xi,\eta)$ must satisfy the Cauchy Riemann relations in the domain R, namely,

$$x_\xi = y_\eta \text{ and } x_\eta = - y_\xi .$$

Consequently, the condition of integrability yields

$$\nabla^2 x = 0 \text{ and } \nabla^2 y = 0.$$

It is clear from these relationships that it is not possible to fix both x and y on the boundary of R for the solution of the Laplacians. One possibility is to fix the sides ξ=constant of the rectangle R with the value of the function of x and from the second Cauchy-Riemann relation the value of the derivative y_ξ, while on the sides η=constant the function y and the derivatives x_η are specified. Hence, it is apparent that in the case of conformal mapping it is not possible to specify the distribution of the grid points because of the constraint placed upon the metrics g_{11} and g_{12}. A slightly more flexible approach can be achieved if the constraint $g_{11}=g_{22}$ is relaxed and the relationship is taken as

$$g_{11} = F^2(\xi,\eta)\, g_{22} \text{ and } g_{12} = 0$$

where the specification of F, a dilatation function, permits a certain flexibility without the loss of orthogonality[22].

As an example of conformal mapping applied to grid generation, consider the particularly attractive method for 2-dimensional aerofoil configurations based upon the Joukowski mapping,

$$z = \zeta + \frac{l^2}{\zeta}$$

A circle in the ζ plane maps to a so-called Joukowski aerofoil in the z plane. A generalisation of this to general aerofoil geometries is the Von Karman - Treffitz mapping

where z is the aerofoil plane complex coordinates, ζ the near-circle plane complex coordinate, $\kappa=2-\lambda/\pi$, with λ the included trailing edge angle, z_{1t} and z_{1n} are two singular points in the z plane at the trailing edge and at the midway between the nose of the aerofoil and its centre of curvature, respectively, and the parameter s_1 is related to z_{1t} and z_{1n} through the relations $z_{1t}= z_{1s} + ks_1$ and $z_{1n} = z_{1s} - ks_1$. A polar mesh with appropriate stretching in the radial direction generated around the near circle, when mapped back to the physical plane, produces a high quality structured mesh. The technique is computationally efficient and has been used to provide a suitable set of points for aerofoil-like geometries. An example of this is shown in Fig. 2.2.8.

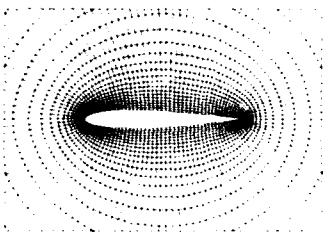


Figure 2.2.8 A structured conformal grid

Conformal mappings have found many early applications in the numerical simulation of potential flow[23]. Although the resulting grids preserve a basic cell shape, the amount of local control provided is not sufficient for many problems. As indicated, one by one, the intrinsic properties of conformal mappings must be dropped to provide greater flexibility. Cell shape preservation can be replaced by only orthogonality, thus allowing the grid to stretch in one or more of the coordinate curve directions. Shearing transformations, in turn, can overcome further shortcomings of the conformal mapping. Extensions to three dimensions can be achieved by conformal mappings in two dimensions followed by an algebraic stretching in the third dimension.

The requirement for more general algebraic grid generation procedures has lead to the investigations of general uni-directional interpolation methods based upon Lagrange and Hermite interpolations, which can then be extended to multi-directional formulations. Shearing transformations, and general interpolants have been studied and effective techniques, such as the multisurface[24-26] and transfinite interpolation[27,28] procedures, have been developed.

Transfinite Interpolation

In its basic form transfinite interpolation can be described initially in turns of one dimensional shearing transformations. In 2 dimensions consider the transformation

$$R_\xi(\xi,\eta) = (1-\xi)R(0,\eta) + \xi R(1,\eta)$$

which expresses the interpolation R_ξ derived by interpolating between the boundaries η=constant. Similarly, for the boundary ξ=constant, the interpolant

$$R_\eta(\xi,\eta) = (1-\eta)R(\xi,0) + \eta R(\xi,1)$$

is obtained. The tensor product of R_ξ and R_η gives an interpolant which maps the four corners of the computational domain to the four corresponding corners of the physical domain. The remaining boundary points between the two domains have no correspondence under this mapping. This occurs because the boundaries map into line segments between corners. By contrast, the simple sum $R_\xi + R_\eta$ maps each boundary to the sum of a line segment and the actual physical boundary. By using the tensor product mapping to remove the boundary line segments, the Boolean sum is obtained

$$R_\xi \oplus R_\eta = R_\xi + R_\eta - R_\xi R_\eta$$

where the tensor product is defined as

$$R_\xi R_\eta = (1-\xi)[(1-\eta)R(0,0)+\eta R(0,1)]+\xi[(1-\eta)R(1,0)+\eta R(1,1)] .$$

This represents transfinite interpolation. In practice, the previous equation is broken down into several components. For example, in three dimensions the interpolant would be expressed as

$$\begin{aligned} F_1 &= R_\zeta [F] \\ F_2 &= F_1 + R_\eta [F - F_1] \\ F_3 &= F_2 + R_\xi [F - F_2] \end{aligned} \qquad (2.2.8)$$

where ζ is the computational variable in the third direction. $F-F_1$ represents the mismatch between the actual surface and the first interpolant.

In the above description, uni-directional interpolants of the Lagrange form have been used. If control of local boundary slopes are required then these interpolants can be expressed in Hermite form to give

$$R_\xi = f(\xi_1,\eta)\,\alpha_1(\xi)+\frac{\partial f}{\partial \xi}\,\alpha_2(\xi)+f(\xi_2,\eta)\alpha_3(\xi)+\frac{\partial f}{\partial \xi}\,\alpha_4(\xi) .$$

Similar expressions can be written for the interpolants R_η and R_ζ. The coefficients α_i, i=1,4 are the blending functions with again corresponding equivalent terms for the η and ζ interpolants.

It should be noted that transfinite interpolation is a direct evaluation of Eq. (2.2.8, and hence is computationally very fast and efficient.

2.2.3 Structured Grids from Variational Methods

As seen in Section 2.2.1, elliptic grid generation through the inverted Laplacian without grid control functions can be obtained by optimizing the smoothness of the distribution of grid points. Based on this observation, different variational techniques have been developed in order to minimize a global functional which amalgamate different grid properties to be optimized[13,30,31,33].

The general form of these variational methods allows a competitive enhancement of grid smoothness, orthogonality and point concentration by representing each of these desired properties by integral measures over the grid and minimizing a weighted average of them.

The scope of the variational methods used for optimization of structured grids is also applicable to the optimization of unstructured grids (Section 2.2.5)[32].

Among the properties considered for mesh optimization, some are quantified either in terms of mapping between the physical space $\mathbf{x}(x,y,z)$ and a uniformly discretized reference space ξ,η,ζ or in terms of mapping between the reference space ξ,η,ζ, and the physical space $\mathbf{x}$.

Belonging to the first type, the global functional σ used by Brackbill and Saltzman[13] measures the grid quality as a combination of smoothness, orthogonality and volume control by the following integral:

$$\sigma=\int\{c_s[\nabla\xi^2+\nabla\eta^2+\nabla\zeta^2] + c_o[(\nabla\xi\nabla\eta)^2+(\nabla\eta\nabla\zeta)^2+(\nabla\xi\nabla\zeta)^2]J^3 + c_w w(\xi,\eta,\zeta)J\}dxdydz$$

where c_s, c_o, c_w are the weight factors chosen in order to stress one of the desired properties, and w is a variable factor used to control the volume of the cells. Belonging to the second type, the method of Carcaillet, Kennon and Dulikravich[30] uses a functional that can be written, in a continuous form, as follows:

$$\sigma=\int\{c_s[k_1(\xi,\eta,\zeta)(x_\xi)^2+k_2(\xi,\eta,\zeta)(x_\eta)^2+k_3(\xi,\eta,\zeta)(x_\zeta)^2] + c_o[(x_\xi.x_\eta)^2+(x_\eta.x_\zeta)^2+(x_\zeta.x_\xi)^2] + c_w\, w(\xi,\eta,\zeta)\,J^2\}\,d\xi\,d\eta\,d\zeta$$

where k_1, k_2, k_3 can be seen as variable stiffness coefficients, in clear analogy with a system where grid points are connected to their immediate neighbours by tension springs.

In the references[33-35] a generalized measure of the mesh deformation between the reference space is presented. The deformation on a non-uniform mesh and the physical space is presented. The deformation characterizes the energy of an hyperelastic, isotropic, homogeneous material satisfying the axiom of frame indifference. In order to establish a well posed minimization problem, the functional is enforced to be locally convex in the neighbourhood of a rigid transformation.

According to these mechanical and mathematical properties, the functional σ, in a continuous form, can be expressed as

$$\sigma=\int\{c_1(I_1-J^2-2)+c_2(I_2-2J^2-1)+c_3(J-1)^2\}d\xi\,d\eta\,d\zeta$$

where

$$I_1=(x_\xi)^2+(x_\eta)^2+(x_\zeta)^2$$

$$I_2=(x_\xi \times x_\eta)^2+(x_\eta \times x_\zeta)^2+(x_\zeta \times x_\xi)^2$$

$$J=(x_\xi \times x_\eta)\cdot x_\zeta$$

Solution algorithm

The basic pattern of development with variational methods consists in defining a pointwise positive measure gathering together some desired grid features, to integrate this measure over the field and to minimize the resulting functional.

Two alternative approaches are commonly used for the solution of the minimization process:

1. The first approach consists in expressing the optimality conditions known as the Euler Lagrange equations giving a system of partial differential equations to be discretized and solved by an appropriate numerical method[13,14].

2. The second approach is simply the direct minimization of the global function σ after its discretization. The most popular algorithms used for such an optimization problem are the Fletcher-Reeves and the Polak-Ribiere conjugate gradient methods. Further discussion of the minimization process an be found elsewhere[30,34].

2.2.4 The Multiblock Approach

The techniques discussed for the generation of structured grids imply a mapping between the physical space, (x,y,z) and the regular transformed space (ξ,η,ζ). This indicates that, whatever the shape of the domain in the physical space, it is topologically equivalent to a cuboid. For general shaped flow domains this is a major restriction, since this can give rise to inappropriate grid structures and, in general, an over constraining of the grid generation equations. At the heart of this problem is the global conservation of the curvilinear coordinates. For application to general shapes this proves too restrictive and it is necessary to introduce a multiblock subdivision of the domain[3-7,36,37].

The idea behind multiblock is that, instead of utilising one global curvilinear coordinate system, several curvilinear systems are constructed and subsequently connected together. The domain is subdivided into regions, each of which is topologically equivalent to a cuboid and within which a structured grid is generated. The block subdivision provides the necessary flexibility to construct structured meshes for complex shapes. The approach represents a compromise between the globally structured grid and an unstructured grid.

The multiblock concept has proved to be powerful in the construction of high quality grids for aerospace geometries. The arrangement of blocks defines how the local curvilinear systems connect and the resulting connectivities between the blocks define the global grid topology. It is possible to construct a wide range of mesh topologies for any given configuration. In particular, it is possible to construct 'component adaptive' mesh topologies to ensure that the mesh lines close to a component are appropriate to the geometrical shape of that component. The multiblock concept is not particular to any grid point generation technique. The generation of points can be achieved using the algebraic or elliptic procedure.

The form of a structured grid is often described by the topological structure. The 3 basic forms are 'O' or polar, 'H' and 'C' grids. To achieve these different topologies, 3 different mappings are required. Figure 2.2.9 a, b and c show the 3 mappings, in 2 dimensions, required for these grid structures. They are shown here in multiblock form, where the convention has been applied that there is only one grid boundary condition type along any edge.

These basic mappings can be used to build different grid structures for any given configuration. For example, consider a 2 aerofoil system, arranged in a tandem configuration. An 'H' grid could be constructed local to each aerofoil in a way which is shown in schematic form in Fig. 2.2.10.

Alternatively, the forward aerofoil could be favoured with a 'C' topology, whilst the grid around the aft component remains of 'H' type. The schematic for this is shown in Fig. 2.2.11. In a more adventurous way, it may be appropriate to construct a C-structure around the leading edge of both aerofoils, as shown in Fig. 2.2.12 or a polar mesh around both aerofoils, as indicated in Fig. 2.2.13. Either of these grid topologies are valid, and the associated block structure for the 2 are shown in Figs. 2.2.12b and 2.2.13b, respectively. The 'C' structure requires 18 blocks whilst the polar mesh requires 36.

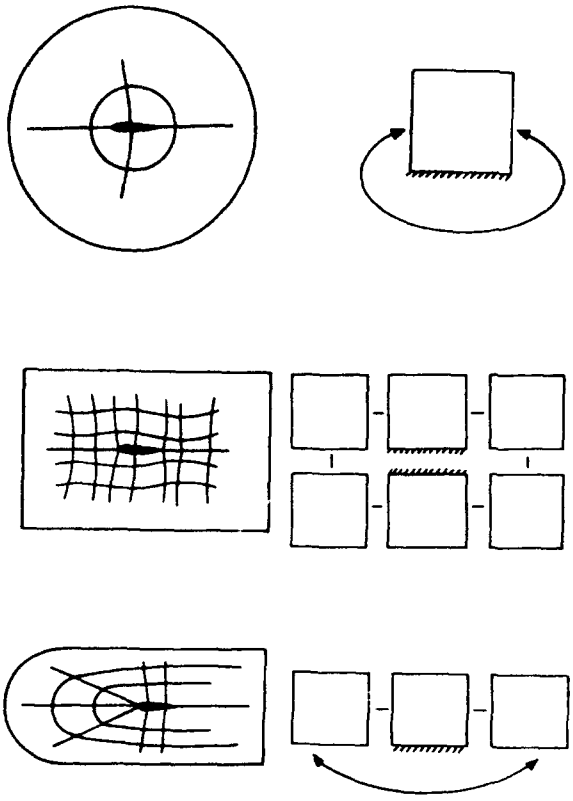


Figure 2.2.9 The basic mappings of structured grids.

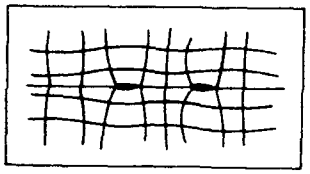


Figure 2.2.10 H-grid topology for a tandem configuration.

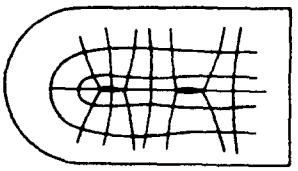


Figure 2.2.11 C-H grid topology for a tandem configuration.

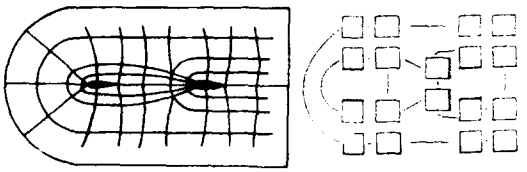


Figure 2.2.12
a) C-C grid topology for a tandem configuration.
b) Block structure for the C-C grid topology

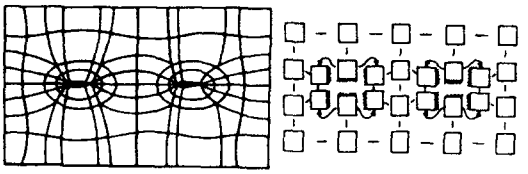


Figure 2.2.13
a)O-O grid topology for a tandem configuration.
b) Block structure for the C-C grid topology

As indicated by this simple configuration, it is possible to construct many different grid topologies for any configuration. The final choice will be made on the grounds of the detailed geometry of the configuration and on flow conditions.

The arrangements of blocks is defined in a way analogous to the connectivity matrix for an unstructured grid. One approach is to specify for each side of every block,
i) The number of points on the side,
ii) The type of boundary condition
and
iii) if appropriate,
a) the adjacent block number
b) the adjacent side of the adjacent block
c) the orientation of the coordinate system of the adjacent block relative to that of the current block.

The procedure for multiblock grid generation begins with a suitable subdivision of the flow domain into regions, each of which is topologically equivalent to a cuboid in that each has 8 corner points, 12 edges and 6 faces. These regions are connected together to form the grid topology. The connections between blocks are specified in a topology or block connectivity matrix. Such connectivity matrices are of the form used to describe element connections in an unstructured grid.

Once defined, the block connectivity matrix is used to derive the grid structure on the surface of a configuration. This is necessary to generate the grid on the component surfaces and clearly the grid structure on the surfaces must be compatible with the structure in the field.

At this stage, no grid generation has been performed, only the connection between blocks. However, once this has been achieved the grid generation procedure can begin.

The grid structure on the surfaces of the configuration is known and points can be generated on edges of the blocks. To ensure continuity of grid lines across block boundaries it is necessary to form the same computational molecules for all points, including points which are in corners, on edges of faces and on block boundaries. This methodology ensures that grid lines are smooth

everywhere, including across block boundaries. This is an attractive procedure since the block subdivision was utilised only to achieve an appropriate grid structure for the geometry under consideration. It would detract from this approach if, as a by-product, it was to produce grid lines which were discontinuous in slope and higher derivatives across block boundaries. The practical implementation of smooth lines across boundary faces requires the use of a halo or pointer system, but is relatively easy to achieve.

The approach has proved to be very popular and successful and many impressive computations in aerospace engineering have been performed[7]. It has some disadvantages which are now widely recognised. Firstly, a criticism is the difficulty of use of such an approach and secondly, its applicability to all aerospace geometries and flow computations. The first difficulty primarily relates to the specification of the block connectivities. This is a difficult task, since it requires expert user effort and, furthermore, if an inappropriate grid topology is defined it is likely to lead to problems late in the grid or flow process. Automatic ways of subdividing a domain have been explored with limited success. Now emphasis appears to be given to interactive specification and grid generation using a graphics workstation[72,73]. However, although this decreases the time for the task it still requires expert user effort and is prone to mistakes. The second problem is the application of the approach to all aerospace geometries. For some configurations, the specification of a suitable block decomposition is difficult and even if achieved can lead to highly constrained grids of poor quality. Much is now understood of the mathematics of structured grid generation and it is likely that grid quality techniques will continue to improve.

2.2.5 Unstructured Grid Methods

Delaunay Triangulation

Dirichlet[38] in 1850 proposed a method whereby a given domain could be systematically decomposed into a set of packed convex polygons. Given two points in the plane, p_i and p_j, the perpendicular bisector of the line joining the two points subdivides the plane into two regions, V_i and V_j. The region V_i is the space closer to p_i than to p_j. Extending these ideas, it is clear that for a given set of points in the plane, the regions V_i are territories which can be assigned to each point such that V_i represents the space closer to p_i than to any other point in the set. This geometrical construction of tiles is known as the Dirichlet tessellation. This tessellation of a closed domain results in a set of non-overlapping convex polygons, called Voronoi regions, covering the entire domain.

A more formal definition can be stated. If a set of points is denoted by $\{p_i\}$, then the Voronoi region $\{V_i\}$ can be defined as

$$\{V_i\} = \{\, p : \|p-p_i\| < \|p-p_j\|, \text{ for all } j \neq i \,\}$$

i.e. the Voronoi region $\{V_i\}$ is the set of all points that are closer to p_j than to any other point. The sum of all points forms a Voronoi polygon.

From this definition, it is apparent, that in two dimensions, the territorial boundary which forms a side of a Voronoi polygon must be midway between the two points which it separates and is thus a segment of the perpendicular bisector of the line joining these two points. If all point pairs which have some segment of boundary in common are joined by straight lines, the result is a triangulation of the convex hull of the set of points $\{p_i\}$. This triangulation is known as the Delaunay triangulation[39]. An example of this construction, illustrated in two dimensions is shown in Fig. 2.2.14.

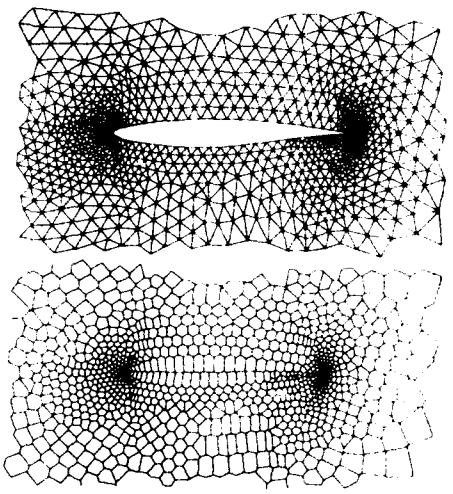


Figure 2.2.14 The Delaunay and Voronoi constructions.

Equivalent constructions can be defined in higher dimensions. In three dimensions, the territorial boundary which forms a face of a Voronoi polyhedron is equidistant between the two points which it separates. If all point pairs which have a common face in the Voronoi construction are connected then a set of tetrahedra is formed which covers the convex hull of the data points.

The Delaunay triangulation has some rather interesting properties[40]. One of particular interest is the so-called in-circle criterion. The vertices of the Voronoi diagram are at the circumcentres of the circles which pass through the three points which forms a triangle. In three dimensions, the Voronoi vertices are at the centre of the sphere which passes though the four points which form a tetrahedron. It follows from the definition of the Dirichlet tessellation that no points, other than the so-called forming points which form the triangles or tetrahedra, fall within the circles or spheres. If a point did fall inside then this would contradict the basic definition. This geometrical property is the in-circle criterion.

This criterion forms the basis for the most popular algorithms for the construction of the tessellation which were proposed by Bowyer[41] and Watson[42].

The basic outline of the algorithm of Bowyer, which is applicable in 3 dimensions is:-

i). Define the convex hull within which all points will lie. It is appropriate to specify 8 points together with the associated Voronoi diagram structure.

ii). Introduce a new point anywhere within the convex hull $\{1,\ldots,8\}$.

iii). Determine all vertices of the Voronoi diagram to be deleted. A point which lies within the sphere, centred at a vertex of the Voronoi diagram and which passes through its four forming points, results in the deletion of that vertex. This follows from the 'in-circle' definition of the Voronoi construction.

iv). Find the forming points of all the deleted Voronoi vertices. These are the contiguous points to the new point.

v). Determine the Voronoi vertices which have not themselves been deleted which are neighbours to the deleted vertices. These data provide the necessary information to enable valid combinations of the contiguous points to be constructed.

vi). Determine the forming points of the new Voronoi vertices. The forming points of new vertices must include the new point together with the three points which are contiguous to the new point and form a face of a neighbouring tetrahedra (these are the possible combinations obtained from Step v).

vii). Determine the neighbouring Voronoi vertices to the new Voronoi vertices. Following Step vi, the forming points of all new vertices have been computed. For each new vertex, perform a search through the forming points of the neighbouring vertices as found in Step v to identify common pairs of forming points. When a common combination occurs, then the three associated vertices are neighbours of the Voronoi diagram.

viii). Reorder the Voronoi diagram data structure, overwriting the entries of the deleted vertices.

ix). Repeat steps (ii-viii) for the next point.

The Delaunay triangulation is a systematic way to connect together an arbitrary set of points in either 2 or 3 dimensions. It does not provide a technique by which the coordinates of points can be generated. In the early applications of the method, grid points were computed using a technique based on structured grid generation[43,44]. A set of grid points was generated for each individual component of a configuration and the resulting collection of all points was then connected to form an unstructured grid. Points which fell inside component boundaries were automatically detected and then rejected. Such an example is shown in Figure 2.2.15.

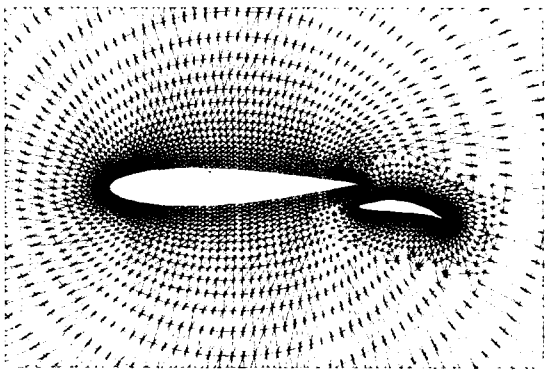


Figure 2.2.15 Delaunay triangulation of a structured set of grid points.

This proved to be successful, but it is a clearly limited approach when applied to arbitrary geometries since it involved the definition of an auxiliary set of points. This motivated the search for automatic ways to generate points within the Delaunay triangulation procedure. Several have now been developed and applied to realistic configurations[45-49].

An example of a procedure to add interior grid points in two dimensions in the Delaunay triangulation is presented[48,49]. The computational domain is defined in discrete form by the boundary points. It will be assumed that this point distribution reflects appropriate geometrical features, such as variation in curvature and gradient. An algorithm which creates points within the domain which reflects the boundary point spacing is as follows:

i) Compute the point distribution function for each boundary point $\mathbf{r}_i = (x,y)$, i.e. for point i

$$dp_i = 0.5(\mathrm{sqrt}(\mathbf{r}_{i+1}-\mathbf{r}_i)^2 + \mathrm{sqrt}(\mathbf{r}_i-\mathbf{r}_{i-1})^2),$$

where it is assumed that points i+1 and i-1 are contiguous to i.

ii) Generate the Delaunay triangulation of the boundary points.

iii) Initialize the number of interior field points created, N=0.

iv)For all triangles within the domain,

a) Define a prospective point, Q, to be at the centroid of the triangle.

b) Derive the point distribution, dp, for the point Q, by interpolating the point distribution function from the nodes of the triangle, dp_m, m=1,2,3.

c) Compute the distances d_m, m=1,2,3 from the prospective point, Q, to each of the points of the triangle.

If $\{d_m < \alpha dp_m\}$ for any m=1,2,3 then reject the point :- Return to the beginning of step (iv).

If $\{d_m > \alpha dp_m\}$ for any m=1,2,3 then

Compute the distance s_j, (j=1, N), from the prospective point Q, to other points to be inserted, P_j, j=1,N.

If $\{s_j < \beta dp_m\}$ then reject the point :- Return to the beginning of step (iv).

If $\{s_j > \beta dp_m\}$ then accept the point Q for insertion by the Delaunay triangulation algorithm. Include Q in the list P_j, j=1,N.

d) Assign the interpolated value of the point distribution function, dp, to the new node, P_N.

e) Next triangle.

v) If N=0 go to step (vii).

vi) Perform Delaunay triangulation of the derived points, P_j, j=1,N. Go to step (iii)

vii) Smooth the mesh.

The coefficient α controls the grid point density, whilst β has an influence on the regularity of the triangulation. The effects of the parameters α and β are demonstrated in the following examples. Figs. 2.2.16a and b, show two triangulations obtained from the boundary points which define the region between 2 concentric circles.

A more realistic example of the automatic point creation algorithm is given in Fig. 2.2.17, where a grid is shown around a multiply connected airfoil system and a value of β=0.02 has been used. It is clear from the examples given that the 'exclusion zone' for point creation, which is a circle of radius βdp_m, has an effect of controlling the regularity of the triangulation.

The method proposed for creating points can be generalised and applied with a background mesh or point and line sources to control grid point spacing. In these cases, the local point distribution function is computed from a background mesh or distance from a source which are either specified by the user or derived from a previous flow computation, if grid adaptivity is applied. Fig. 2.2.18 shows some examples of these techniques.

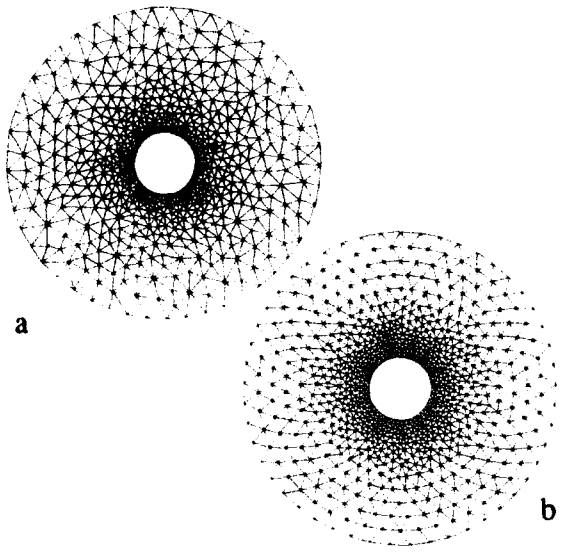


Figure 2.2.16
a) Automatic point creation α=1.0, β=10.0
b) Automatic point creation α=1.0, β=0.10

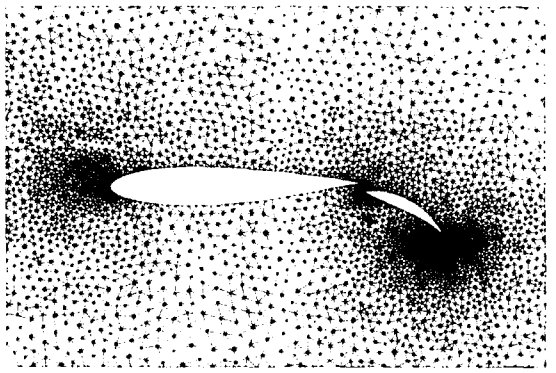


Figure 2.2.17 Multi-component aerofoil system.

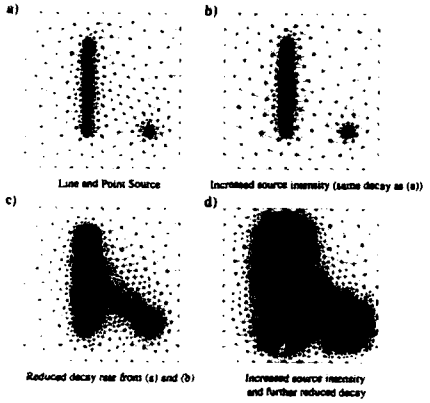


Figure 2.2.18 Examples of the use of point and line sources in the automatic point creation routine.

The method also extends with complete generality into three dimensions[48,49].

As was briefly discussed in the introductory comments, one of the basic requirements of the grid is for it to be boundary conforming. In the Delaunay triangulation, boundary conformity has to be checked and if necessary enforced by the use of special techniques. Given a set of points which describe a geometry in 2 or 3 dimensions and the Delaunay connections between these points there is no guarantee that the resulting triangulation will contain edges or faces which conform to the boundary surface. In fact, for complicated shapes, the boundary edges and faces will almost certainly not be recovered. The techniques devised to correct the triangulation or force boundary integrity are many. In 2 dimensions, the given boundary edges can be recovered by edge swapping and does not really represent a significant problem. In 3 dimensions the problem is much more severe. Baker[50] chooses to introduce skeleton points to ensure boundary integrity whilst George[51] and Weatherill[48,49] have chosen to perform tetrahedral transformations to recover boundary faces. This problem is often overlooked in favour of the triangulation algorithm. However, the construction of the algorithm to form the connections is very well defined and can be relatively easily programmed. Boundary integrity is a less well defined problem and as such is more problematic to solve.

Advancing Front

A grid generation technique which is based upon the simultaneous point generation and connection is the advancing front method. Given a set of points which defines a geometrical boundary or boundaries and a measure of the local spacing required within the domain, the method extends or advances the boundary connectivity into the field. Grid points are generated and connected to other local points and in this way the grid is advanced away from the boundaries. The grid point density is controlled by the user specified parameters which is often called the background mesh. For a uniform distribution of points, the background mesh can be a single triangle with point spacing parameters assigned to each of the three nodes. Points in the interior of the domain are created to be consistent with the background spacing.

The nature of the method makes it ideally suited to complicated geometries and requires a minimal input from the user. The approach, first discussed by George[52] has been enhanced and extended into 3 dimensions by Peraire et al.[53,54] and Lohner[55].

The approach of Peraire et al. is to define a mesh parameter δ which controls the local node spacing and a local stretching parameter $\underline{\alpha}$ and the local direction $\mathbf{t}$ of stretching. These parameters allow a grid to be generated with variable sized elements and which are clustered and stretched in such a way that one-dimensional features in a solution can be captured in a very efficient manner.

Before starting to generate triangles inside a domain, the body points are created and the initial front, containing the information about all the boundary sides, is initialized. The orientation of the boundary contours is such that the interior of the domain is always to the left. The initial front is defined by two integer vectors which have as many components as actual active sides. In one vector the number of the first node of each side is stored whereas the second node number is kept in the other vector. The generation front changes continuously throughout the process of triangulation and must be updated whenever a new element is formed. Thus the front changes during the triangulation process and reduces to zero when the end of the triangulation is reached. Every time a triangle is generated its sides are added as new entries in the front.

In the process of generating a new triangle, the following steps are involved.
1) A front side, with nodes A and B, to be used as the base for a triangle is selected. With distance δ_1 from points A and B, a

point C is chosen. δ_1 is chosen as

$$\delta_1 = \begin{cases} \delta & \text{if } 0.55*AB < \delta < 2*AB \\ 0.55*AB & \text{if } 0.55*AB > \delta \\ 2*AB & \text{if } \delta > 2*AB \end{cases}$$

2) Find all the nodes which belong to any of the current front sides and lie inside the area determined by the circle (centre C and radius r=3*AB) and the points A and B. Order nodes (a_i, i=1,n) according to their distance from the point C, in such a way that the first node in the list is the closest to C.

3) Decide whether or not point C is added to the list. If

$$Aa_1 < 1.5*\delta_1$$

$$Ba_1 < 1.5*\delta_1$$

then C is not included. Otherwise C is placed in the first position of the list and the other nodes are shifted by one place.

4)Determine the connecting point a_j. This is taken to be the first node in the list that satisfies the two following validity requirements.

i) the interior of the triangle ABa_j does not contain any of the remaining nodes in the list (excluding C).

ii) The segment line a_jM (M midpoint of AB) does not intersect any of the existing sides of the front.

5) Form and store the new element. If C is chosen as the connecting point a new node is created.

This approach can be extended to a non-uniform distribution of δ and also the inclusion of element stretching.

2.2.6 Other Grid Types

Overlaid grids

The basic techniques outlined indicate that the generation of a grid around a simple geometrical shape is not too problematic. In fact, sufficient is now known of the mathematics of grid generation that high quality, controlled grids can be generated. Only when the geometry becomes complicated is it necessary to resort to more complex strategies, such as multiblock. The idea that any complicated shape can be broken into geometrically simple components, each of which can be gridded, is at the heart of the overlaid grid approach.

The overlaid approach generates a suitable grid around components of a configuration and each grid is allowed to overlap other grids. Interpolation of flow data is then used in the region of overlap. This approach, which has been taken to a considerable degree of sophistication is often called the 'Chimera' method[18,19]. The key problem in the approach is the accurate and conservative transfer of data between component grids. The approach has been used with considerable success in simulating moving boundary problems.

Hybrid grids

The hybrid grid approach is similar to the overlaid grid method in that a structured grid is used wherever is appropriate, but where the geometry becomes complicated an unstructured grid is used[56-59]. Both grid types are then connected together in a node-to node continuous manner. The hybrid approach can be utilised with a multiblock approach. A multiblock grid is generated around a part of the configuration, with the additional components gridded with an unstructured grid. The unstructured grid can be generated using either the advancing front or Delaunay technique. Fig. 2.2.19 shows a typical example of a hybrid grid in 2 dimensions. Fig. 2.2.20 shows a typical example in 3 dimensions, where the nacelle and pylon have been discretized with an unstructured grid which has then been connected to a structured multiblock grid.

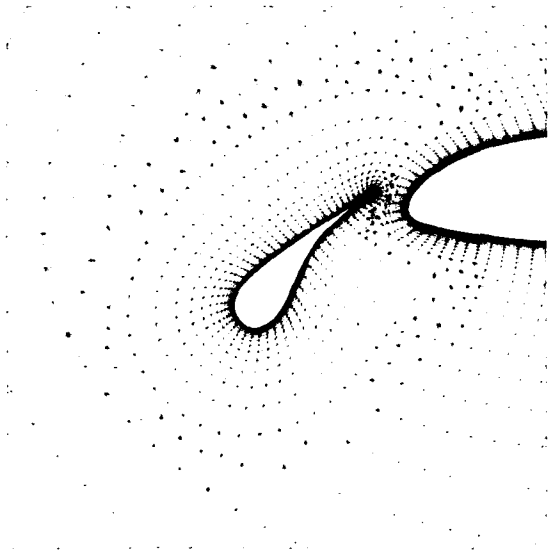


Figure 2.2.19 A hybrid grid in 2 dimensions

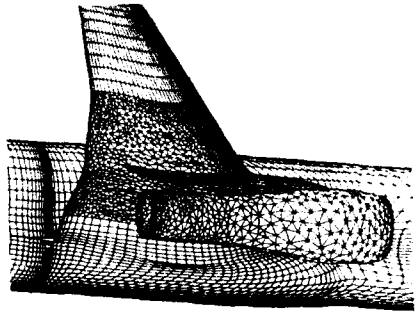


Figure 2.2.20 An example of a hybrid grid in 3 dimensions.

2.2.7 Surface Grids

Surface mesh generation is, in itself, one of the most difficult and yet important aspects of mesh generation in 3 dimensions. The surface mesh influences the field mesh close to the boundary, which is often where rapid changes in flow variables take place. Surface meshes have the same requirement for smoothness and continuity as the field meshes for which they act as boundary conditions, but in addition, they are required to conform to the configuration surfaces, including lines of component intersection and to adequately model regions of high surface curvature. Several approaches have been suggested in the literature, but most rely upon a parametric representation of a surface.

A suitable numerical representation of a surface involves aspects of geometry modelling[60,61]. Geometry modelling is a means by which a continuous surface can be defined from a discrete set of points. Such a description of a surface is valuable for the generation of surface meshes which, in general, will not coincide with the original geometry definition. A parametric representation of a surface is straightforward to construct and provides a description of a surface in terms of two parametric coordinates. This is of particular importance, since the generation of a mesh on a surface then involves using grid generation techniques, developed for 2 space dimensions, in the parametric coordinates.

In this way, surface mesh generation can be viewed as a transformation of the Euclidian Space, $\Re^3$ to a parametric representation in coordinates (s,t) within which the grid is generated to be subsequently mapped back to $\Re^3$ to give the surface mesh. Clearly, a key aspect of this procedure is the mapping between the Euclidian space and the parametric space.

Many such transformations are available. One classic formulation is a due to Ferguson and can be expressed as

$$r(s,t) = F(s)\, Q^T\, F(t)$$

where

$$F(s) = [1 \;\; s \;\; s^2 \;\; s^3] \begin{vmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -3 & 3 & -2 & -1 \\ 2 & -2 & 1 & 1 \end{vmatrix}$$

and

$$Q = \begin{vmatrix} r(0,0) & r(0,1) & r_t(0,0) & r_t(0,1) \\ r(1,0) & r(1,1) & r_t(1,0) & r_t(1,1) \\ r_s(0,0) & r_s(0,1) & r_{st}(0,0) & r_{st}(0,1) \\ r_s(1,0) & r_s(1,1) & r_{st}(1,0) & r_{st}(1,1) \end{vmatrix}$$

The matrix Q involves coordinates, derivatives in s and t, and cross, or twist derivatives in s and t. The surface is represented by a set of quadrilateral patches and within each quadrilateral patch the Ferguson representation is applicable. Hence any surface which is defined in terms of a network of lines which form quadrilateral patches can be expressed in parametric coordinates (s,t). The grid generation is performed in the coordinates (s,t) before using the Ferguson formulation to convert to physical coordinates. Fig. 2.2.21 shows the sequence involved in the process to generate a structured grid on the surface of a configuration.

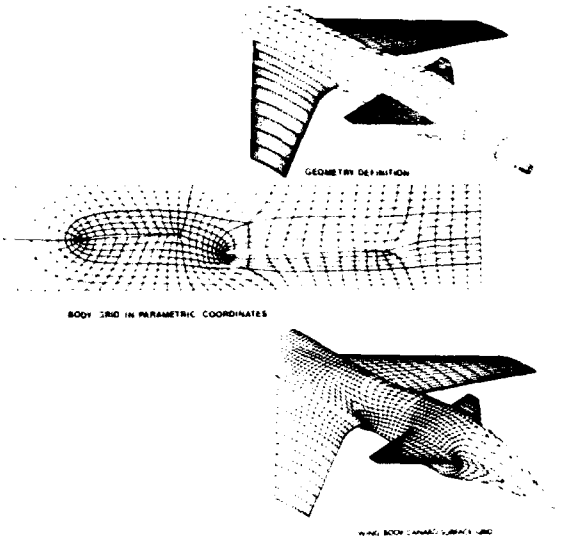


Figure 2.2.21 The steps in the surface grid generation.

A similar procedure is followed to generate an unstructured grid on a surface.

Surface-to-surface intersections

One aspect relating to surface grid generation which is often encountered is the computation of surface-to-surface intersections. Obviously, it is necessary to place grid points on such intersection curves.

If two surfaces intersect and they are both represented in parametric form as just described then it is possible to formulate an equation which defines the intersection curve.

Assume the two surfaces $\mathbf{x}^1$ and $\mathbf{x}^2$ are represented by

$$\mathbf{x}^1 = (x,y,z)^1 = [\, s^3 \; s^2 \; s \; 1\,]\, M_i \begin{vmatrix} t^3 \\ t^2 \\ t \\ 1 \end{vmatrix} \qquad i=x,y,z$$

and

$$\mathbf{x}^2 = (x,y,z)^2 = [\, u^3 \; u^2 \; u \; 1\,]\, N_i \begin{vmatrix} v^3 \\ v^2 \\ v \\ 1 \end{vmatrix} \qquad i=x,y,z$$

The (s,t) and (u,v) are sets of parametric coordinates and M_i and N_i matrices containing the blending functions and parametric derivatives of $\mathbf{x}$. For an intersection for a constant parameter, say v_o, the following equation must be satisfied

$$[\, s^3 \; s^2 \; s \; 1\,]\, M_i \begin{vmatrix} t^3 \\ t^2 \\ t \\ 1 \end{vmatrix} - [\, u^3 \; u^2 \; u \; 1\,] N_i \begin{vmatrix} v_o^3 \\ v_o^2 \\ v_o \\ 1 \end{vmatrix} = 0$$

This results in three non-linear algebraic equations for s, t and u which can be solved using the Newton-Raphson method. Having solved, it is then possible to convert these parametric coordinates to obtain the position of the intersection in the physical space, $\mathbf{x} = (x,y,z)$.

2.2.8 Grid Adaptivity Techniques

To resolve features of a flowfield accurately it is, in general, necessary to introduce grid adaptivity techniques[62-71]. Adaptivity is based upon the equidistribution of errors principle, namely,

$$w_i \, ds_i = \text{constant}$$

where w_i is the error or activity indicator at node i and ds_i is the local grid point spacing at node i.

Central to adaptivity techniques, and the satisfaction of this equidistribution principle, is to define an appropriate indicator w_i. Adaptivity criteria are based upon an assessment of the error in the solution of the flow equations or are constructed to detect features of the flowfield. These estimators are intimately connected to the flow equations to be solved. For example, some of the main features of a solution of the Euler equations can be shock waves, stagnation points and vortices, and any indicator should accurately identify these flow characteristics. However, for the Navier-Stokes equations, it is important not only to refine the mesh in order to capture these features but, in addition, to adequately resolve viscous dominated phenomena, such as the boundary layers. Hence it seems likely, that certainly in the near future, adaptivity criteria will be a combination of measures each dependent upon some aspects of the flow and, in turn, upon the flow equations.

Many different physical criteria have been suggested for use with the Euler equations. Such measures include,

$$|u.\nabla s|, \; |u.\nabla u|, \; |u.\nabla \rho|, \; |u.\nabla p|, \; |\nabla M|, \; |\nabla \rho|, \; |\nabla p|$$

where s, u, ρ, p, M are entropy, velocity, density, pressure and Mach number, respectively.

There is also an extensive choice of criteria based upon error analysis. Such measures include, a comparison of computational stencils of different orders of magnitude, comparison of the same derivatives on different meshes e.g. Richardson's extrapolation and resort to classical error estimation theory. No generally applicable theory exists for errors associated with hyperbolic equations, hence, to date combinations of rather adhoc methods have been used.

It is important that the adaptivity criteria resolve both the discontinuous features of the solution (i.e. shock waves, contacts) and the smooth features as the number of grid points are increased. A desirable feature of any adaptive method to ensure convergence is that the local cell size goes to zero in the limit of an infinite number of mesh points[70].

Once an adaptivity criterion has been established the equidistribution principle is achieved through a variety of methods, including point enrichment, point derefinement, node movement and remeshing, or combinations of these.

Grid refinement

Grid refinement, or h-refinement, involves the addition of points into regions where adaptivity is required. Such a procedure clearly provides additional resolution at the expense of increasing the number of points in the computation.

Grid refinement on unstructured grids is readily implemented. The addition of a point or points involves a local reconnection of the elements, and the resulting grid has the same form as the initial grid. Hence, the same flow solver can be used on the enriched grid as was used on the initial grid. An example of this approach is given in Fig. 2.2.22.

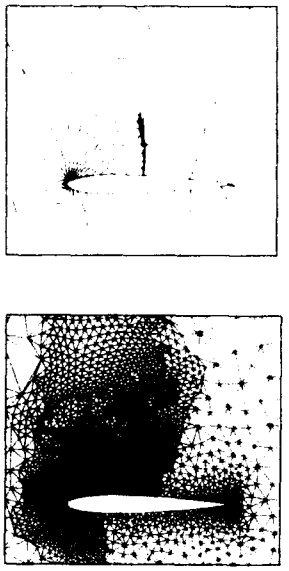


Figure 2.2.22 H-refinement on an unstructured grid.

Grid refinement on structured or multiblock grids is not so straightforward The addition of points will, in general, break the regular array of points. The resulting distributed grid points no longer naturally fit into the elements of an array. Furthermore, some points will not 'conform' to the grid in that they have a different number of connections to other points. Hence grid refinement on structured grids requires a modification to the basic data structure and also the existence of so-called non-conforming nodes requires modifications to the flow solver. Clearly point enrichment on structured grids is not as natural a process as the method applied on unstructured grids and hence is not so widely employed. Work has been undertaken to implement point enrichment on structured grids and the results demonstrate the benefits to be gained from the additional effort in modifications to the data structure and flow solver. Fig. 2.2.23 shows point enrichment on a multiblock structured grid together with the flow contours of pressure and Fig. 2.2.24 shows grid refinement on a single block structured grid.

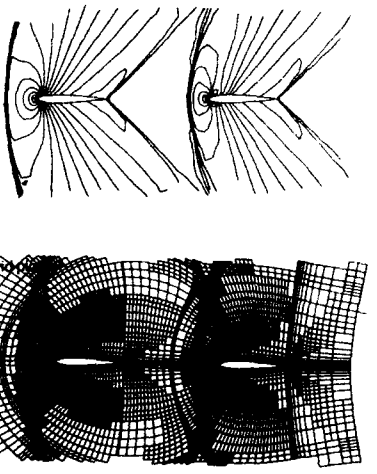


Figure 2.2.23 H-refinement on a multiblock grid

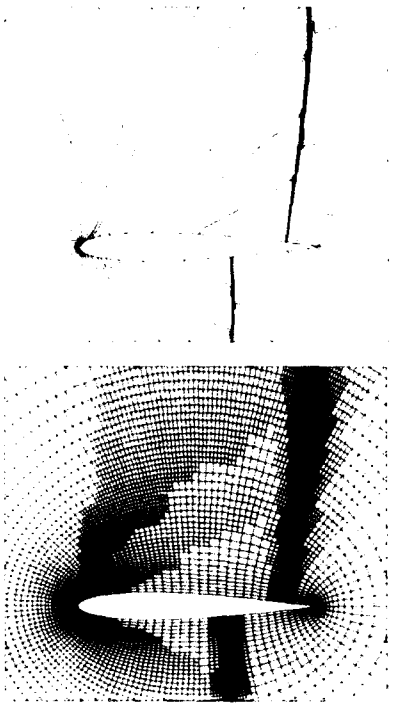


Figure 2.2.24 H-refinement on a single block structured grid

Grid Movement

Grid movement satisfies the equidistribution principle through the migration of points from regions of low activity into regions of high activity. The number of nodes in this case remains fixed. Traditionally, algorithms to move points involve some optimization principle[13]. Typically, as indicated in Section 2.2.3 expressions for smoothness, orthogonality and weighting according to the flowfield or errors are constructed and then an optimization is performed such that movement can be driven by a weight function, but not at the expense of loss of smoothness and orthogonality.

An alternative approach is to use a weighted Laplacian function. Such a formulation is often used to smooth grids, and of course the formal version of the formulation is used as the elliptic grid generator presented earlier, Eq. (2.2.1). Written in a general form for both structured and unstructured grids the weighted Laplacian is

$$r_0^{n+1} = r_0^n + \omega \frac{\sum_{i=1}^{M} C_{i0}\,(r_i^n - r_0^n)}{\sum_{i=1}^{M} C_{i0}}$$

where $\mathbf{r}=(x,y)$, r_0^{n+1} is the position of node 0 at relaxation level n+1, C_{i0} is the adaptive weight function between nodes i and 0 and ω is the relaxation parameter. The summation is taken over all edges connecting point 0 to i, where it is taken that there are M surrounding nodes. In practice, this relaxation is typically applied over 50 cycles with a relaxation parameter of 0.1. The weight function C_{i0} can be taken as a measure of any flow parameter such as pressure, density or a measure of local error. This approach proves trivial to implement on all mesh types, but yet its effects are impressive. Fig. 2.2.25 shows a demonstration of the method applied to both a structured and unstructured mesh, respectively.

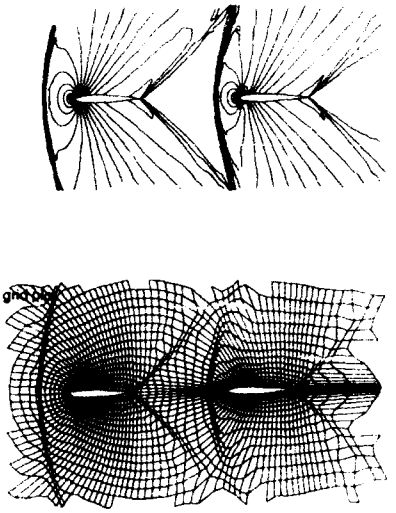


Figure 2.2.25 Node movement

Combinations of Node Movement, Point Enrichment and Derefinement

An optimum approach to adaptation is to combine node movement and point enrichment with derefinement[71]. These procedures should be implemented in a dynamic way, i.e. applied at regular intervals within the flow simulation. Such an approach also provides the possibility of using movement and enrichment to independently capture different features of the flow. An example of combinations of these adaptive mechanisms is given in Fig. 2.2.26.

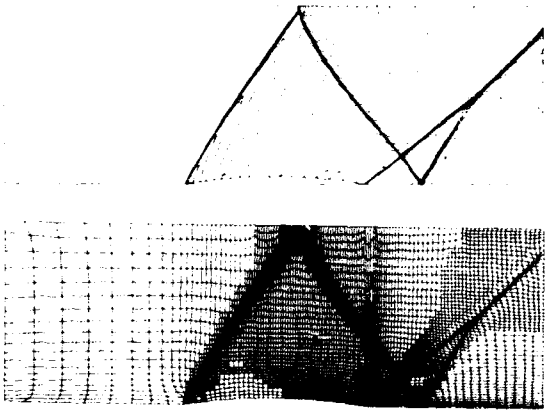


Figure 2.2.26. Combinations of adaptive strategies

Grid Remeshing

One method of adaptation which, to date has been primarily used on unstructured grids, is adaptive remeshing[65]. As already indicated, unstructured meshes can be generated using the concept of a background mesh. For an initial mesh this is usually some very coarse triangulation which covers the domain and on which the spatial distribution is consistent with the given geometry. For adaptive remeshing, the flowfield achieved on an initial mesh is used to define the local point spacing on the background mesh which was itself the initial mesh used for the flow simulation. The mesh is regenerated using the new point spacing on the background mesh. Such an approach can result in a second adapted mesh which contains fewer points than that contained in the initial mesh. However, there is the overhead of regeneration of the mesh which in 3 dimensions can be considerable. Never-the-less impressive demonstrations of its use have been published and such an example is shown in Fig. 2.2.27.

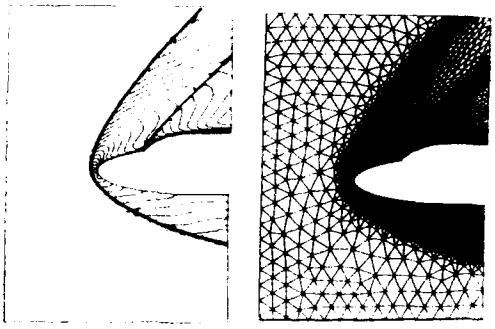


Figure 2.2.27 Grid remeshing for inviscid flows

2.2.9 References

1. SMITH (Ed.) R. E., *Numerical Grid Generation*, NASA Conference Publication, CP-2166, 1980.

2. EISEMAN P. R., "Grid Generation for Fluid Mechanics Computation", Annual Review of Fluid Mechanics, Vol. 17, pp487-522, 1985.

3. HAUSER J. and TAYLOR C. (Eds), *Numerical Grid Generation*, Pineridge Press, Swansea, 1986.

4. THOMPSON J. F. and STEGER J.F (Eds.) *Three Dimensional Grid Generation for Complex Configurations - Recent Progress*, AGARD-AG-309, 1988.

5. SENGUPTA, S.(Eds.), *Numerical Grid Generation in Computational Fluid Dynamics*, Pineridge Press, Swansea, 1988.

6. ARCILLA, A.S., HAUSER, J. EISEMAN, P.R. and THOMPSON J. F. (Eds.) *Numerical Grid Generation in Computational Fluid Dynamics and Related Fields*. North-Holland, Amsterdam, 1991.

7. *Applications of Mesh Generation to Complex 3-D Configurations*, AGARD Conference Proceedings No. 464, May 1989.

8. CLARKE D. K., SALAS M. D. and HASSAN H. A. "Euler Calculations for Multielement Airfoils using Cartesian Grids", AIAA Journal, Vol. 24 (1987), No. 3, pp353-358.

9. DE ZEEUW D. and POWELL K. G. "An Adaptively Refined Cartesian Mesh Solver for the Euler Equations", AIAA Paper 91-1542 (1991).

10. CHIANG Y-l, VAN LEER B. and POWELL K. G. "Simulation of Unsteady Inviscid Flow on an Adaptively Refined Cartesian Grid", AIAA Paper 92-0443 (1992).

11. BERGER M. J. and LEVEQUE R. L. "An Adaptive Cartesian Mesh Algorithm for the Euler Equations in Arbitrary Geometries", AIAA Paper 89-1930 (1989).

12. THOMPSON J. F., THAMES, F. and MASTIN W.,"Automatic Numerical Grid Generation of Body-fitted Curvilinear Coordinate System of Field Containing any Number of Arbitrary 2-Dimensional Bodies", J. Comput. Phys, 15:299-319, 1974.

13. BRACKBILL J. U. and SALTZMAN J. S "Adaptive Zoning for Singular Problems in Two Dimensions", J. Comp. Phys., 46,342, 1982.

14. THOMPSON J. F. , WARZI Z. U. A., MASTIN C. W. , *Numerical Grid Generation: Foundations and Applications*, North-Holland, 1985.

15. THOMAS P. D. and MIDDLECOFF J. F."Direct Control of the Grid Point Distribution in Meshes Generated by Elliptic Equations", AIAA J. 18:652-656, 1980.

16. BOERSTOEL J. W. "Numerical Grid Generation in 3-Dimensional Euler Flow Simulation', In Numerical Methods in Fluid Dynamics", (Eds) K. W. Morton and M. J. Baines, Oxford University Press, 1988.

17. STEGER J. F. and RIZH Y. M. "Generation of Three-Dimensional Body-Fitted Coordinates using Hyperbolic Partial Differential Equations", NASA TM-86753, June 1985.

18. DOUGHERTY F. C. BENEK J. A and STEGER J. L. "On the Application of a Chimera Grid Scheme to Store Separation", NASA TM-88193, 1985.

19. BENEK J. A., BUNING, P.G. and STEGER J. L. "A 3-D Chimera Grid Embedding Technique", AIAA Paper 84-0164, 1984.

20. NAKAMURA S. "Marching Grid Equations using Parabolic Partial Differential Equations", in Numerical Grid Generation', J. F. Thompson(Ed.) North-Holland, Amsterdam, Holland.

21. KOBER H. *Dictionary of Conformal Representation*, Dover, New York, USA.

22. ARINA R. "Orthogonal Grids with Adaptive Control", Numerical Grid Generation in CFD, Ed. Hauser, J. and Taylor C. Pineridge Press, Swansea, 1986.

23. HALSEY N. D. "Potential Flow Analysis of Multielement Airfoils using Conformal Mappings", AIAA J, 17, 1281-1288, 1979.

24. EISEMAN P. R. "A Control Point Form of Algebraic Grid Generation", Int. J. Num. Mthds in Fluids, Vol. 8, pp1165-1181, 1988.

25. EISEMAN P. R. 'A Multi-Surface Method of Coordinate Generation", J. Comp. Phys, Vol. 33, No. 1, pp118-150, 1979.

26. EISEMAN P. R. and SMITH R. E. "Applications of Algebraic Grid Generation" in Applications of Mesh Generation to Complex 3-D Configurations", AGARD Conference Proceedings No. 464, May 1989.

27. GORDON W. N. and HALL C. A. "Construction of Curvilinear Coordinate Systems and Applications to Mesh Generation", Int. J. Num. Mthds in Eng, Vol. 7, pp461-477, 1973.

28. ERIKSSON, L-E. "Generation of Boundary-Conforming Grids around Wing-Body Configurations using Transfinite Interpolation", AIAA Journal, Vol. 20, No. 10, p1313-1320, 1982.

29. EISEMAN, P., "Adaptive Grid Generation", Computer Methods in Applied Mechanics and Engineering, Vol. 64, No. 1-3, pp321-376, 1987.

30. CARCAILLET R., KENNON S. R. and DULIKRAVICH G. S., "Optimization of Three-Dimensional Computational Grids", AIAA Paper 85-4087, 1985.

31. CARCAILLET R., "Optimization of Three Dimensional Computational Grids and Generation of Flow Adaptive Computational Grids", AIAA Paper 86-1058, 1986.

32. CABELLO J., LOHNER R. and JACQUOTTE O. P. "A Variational Method for the Optimization of Two- and Three-Dimensional Unstructured Meshes", ONERA TP 1992-24.

33. JACQUOTTE O. P. and CABELLO J. "Three-dimensional Grid Generation Method based on a Variational Principle", La Recherche Aerospatiale, English Ed. No. 1990-4.

34. JACQUOTTE O. P. "Recent Progress on Mesh Optimization", Proc. 3rd Int. Conf. on Num. Grid Generation in CFD, pp887-898, eds A. S. Arcilla et al., North Holland, Amsterdam, 1991.

35. JACQUOTTE O.P. and COUSSEMENT G.,"Structured Grid Variational Adaption: Reaching the Limit", CFD'92, Eds Ch. Hirsch, J. Periaux and W. Kordulla. Elsevier, Proceedings of the 1st European Computational Fluid Dynamics Conference, Brussels, Belgium, Sept. 1992.

36. LEE, K. D. "3D Transonic Flow Computations using Grid Systems with Block Structure', AIAA Paper 81-0998, 1981.

37. WEATHERILL N. P. and FORSEY C. R. "Grid Generation and Flow Calculations for Aircraft Geometries", J. of Aircraft, Vol. 22, N. 10, p855-860, October 1985.

38. DIRICHLET, G. L. "Uber die Reduktion der Positiven Quadratischen Formen mit drei Unterbestimmten Ganzen Zahlen", Z. Reine Angew. Math., 40,No. 3, pp209-227, 1850.

39. DELAUNAY B. "Sur la Sphere Vide", Bull. Acad. Sci. URSS, Class. Sci. Nat., pp793-800, 1934.

40. PREPARATE F. P. and SHAMOS, M. I. *Computational geometry, an introduction*, Springer-Verlag, 1985.

41. BOWYER A. "Computing Dirichlet tessellations", The Computer Journal, Vol. 24,No. 2, p162-166, 1981.

42. WATSON D. F. "Computing the n-Dimensional Delaunay Tessallation with Application to Voronoi Polytopes", Computer Journal, Vol.24, No.2, pp167-172, 1981.

43. WEATHERILL N. P. "A Method for Generating Irregular Computational Grids in Multiply Connected Planar Domains", Int. J. Numer. methods Fluids, 8, p181-197, 1988.

44. JAMESON A., BAKER T. J. and WEATHERILL N. P. "Calculation of Inviscid Transonic Flow Over a Complete Aircraft", AIAA Paper 86-0103, 1986.

45. BAKER, T. J. "Three-Dimensional Mesh Generation by Triangulation of Arbitrary Point Sets", Proceedings of the AIAA 8th CFD Conference, Hawaii, June 1987, AIAA Paper-87-1124-CP.

46. GEORGE P. L. and HERMELINE F. "Delaunay's Mesh of a Convex Polyhedron in Dimension d; Application for Arbitrary Polyhedra", Int. J. Num. Methd in Eng., vol 33, 975-995 , 1992.

47. SCHROEDER W. J. and SHEPHARD M. S. "Geometry-based Fully Automatic Mesh Generation and the Delaunay Triangulation", Int. J. Numer. Methods Eng., 26, p2503-2515, 1988.

48. WEATHERILL N. P. and HASSAN O. "Efficient Three-Dimensional Grid Generation using the Delaunay Triangulation", Proceedings of the 1st CFD Conference, Brussels, Sept. 1992. Pub. Elsevier, Ed. Ch. Hirsch.

49. WEATHERILL N. P., HASSAN O. and MARCUM D. L."Calculation of Steady Compressible Flowfields with the Finite Element Method", AIAA-93-0341.

50. BAKER T. J. "Shape Reconstruction and Volume Meshing for Complex Solids", Int. J. Num. Mthds in Eng., Vol.32 No. 4, pp665-677, 1991.

51. GEORGE P. L. and HERMELINE F. "Delaunay's Mesh of a Convex Polyhedra in Dimensions d; Application for Arbitrary Polyhedra", Int. J. Num. Methods in Eng., Vol. 33, 975-995 (1992).

52. GEORGE A. J. "Computer Implementation of the Finite Element Method", Stanford University, Dept. of Computer Science, Stan-CS-71-208, Feb. 1971.

53. PERAIRE, J. "A Finite Element Method for Convection Dominated Flows", Ph.D. Thesis, University of Wales, 1986.

54. PERAIRE, J., PEIRO J., FORMAGGIA, L., MORGAN, K. and ZIENKIEWICZ, O. C. "Finite Element Euler Computations in Three Dimensions", Int. J. Num. Meths. Eng, Vol. 26, p2135-2159, 1988.

55. LOHNER R. and PARIKH P. "Three-Dimensional Grid Generation by the Advancing Front Method", Int. J. Num. Meth. Fluids, No. 8, pp1135-1149, 1988.

56. NAKAHASHI K. and OBAYASHI S. "FDM-FEM Zonal Approach for Viscous Flow Computations over Multiple Bodies", AIAA Paper 87-0604, 1987.

57. WEATHERILL N. P. "Mixed Structured-Unstructured Meshes for Aerodynamic Flow Simulation", The Aeronautical Journal, Vol. 94, No. 934, pp111-123, April 1990.

58. WEATHERILL N. P. and NATAKUSUMAH D. K. "The Simulation of Potential Flow around Multiple Bodies using Overlapping Connected Meshes', Applied Mathematics and Computation, 46, pp1-21, 1991.

59. SHAW J. A., GEORGALA J. M., PEACE A. J. and CHILDS P.N. "The Construction, Application and Interpretation of Three-Dimensional Hybrid Grids". Proc. 3rd Int. Conf. on Num. Grid Generation in CFD, pp887-898, eds A. S. Arcilla et al., North Holland, Amsterdam, 1991.

60. COONS S. A."Surfaces for Computer Aided Design of Space Forms", MIT Report MAC-TR-41, 1967.

61. FAUX I. D. and PRATT M. J. *Computational geometry for design and manufacture*. Ellis Horwood, Chichester, 1985.

62. SHEPHARD M. S. and WEATHERILL N. P. (Eds.) "Adaptive Meshing", Special Issue of Int. J. Num. Mthds in Eng., Vol.32 No. 4, 1991.

63. THOMPSON J. F. "Review on the State of the Art of Adaptive Grids", AIAA Paper 84-1606, 1984.

64. DANNENHOFFER J. F. "A Comparison of Adaptive-Grid Redistribution and Embedding for Steady Transonic Flows", Int. J. Num. Mthds in Eng., Vol.32 No. 4, pp651-653, 1991.

65. PERAIRE J., VAHDATI, M. MORGAN K. and ZIENKIEWICZ O.C. "Adaptive Remeshing for Compressible Flow Computations", J. Comput. Physics, 72: 449-466, 1897.

66. HYUN JIN KIM and THOMPSON J. F. "Three Dimensional Adaptive Grid Generation on a Composite Block Grid", AIAA Paper 88-0311, 1988.

67. CATHERALL D. "The Adaption of Structured Grids to Numerical Solutions for Transonic flow", Int. J. Num. Mthds in Eng., Vol.32 No. 4, pp921-939, 1991.

68. NAKAHASHI K. and DEIWERT G. S. "A Practical Adaptive-Grid Method for Complex Fluid-Flow Problems", NASA TM 85989, June 1984.

69. WEATHERILL N.P., HASSAN, O., MARCHANT M. J. and D. L. MARCUM,"Adaptive Inviscid Flow Solutions for Aerospace Geometries on Efficiently Generated Unstructured Tetrahedral Meshes", AIAA Paper 93-3390.

70. WARREN G. P., ANDERSON W. K., THOMAS J.L, and KRIST S. L. "Grid Convergence for Adaptive Methods", AIAA 91-1592.

71. MARCHANT M. J. and WEATHERILL N. P. "Adaptivity Techniques for Compressible Inviscid Flows", Computer Methods in Applied Mechanics and Engineering, 106 (1993) 83-106.

72. SEIBERT W. "An Approach to the Interactive Generation of Block Structured Volume Grids using Computer Graphic Devices", Numerical Grid Generation in CFD, Eds. Hauser and Taylor, Proc. of the 1st Int. Conf., Landshut, FRG, 1986.

73. SEIBERT W., FRITZ W. and LEICHER S. "On the way to an Integrated Mesh Generation System for Industrial Applications", AGARD-CP 464 (1989), Paper 14.

2.3 SPACE DISCRETIZATION - DISSIPATION

All numerical schemes used for obtaining solutions to the Euler equations must contain a certain level of dissipation to prevent odd-even point decoupling, to maintain stability at discontinuities, and to eliminate nonphysical solutions such as expansion shocks. The dissipation may be explicitly added on top of a naturally nondissipative scheme such as pure central differencing, or it may arise naturally from the spatial discretization such as occurs with upwind differencing algorithms. The exact form of dissipation has a large impact on the accuracy of the scheme, as well as the stability and robustness of the overall algorithm. While some discretizations provide some dissipation due to the coupling of the space-time discretizations, these schemes generally have a steady state that depends on the time step and are discussed elsewhere. Below, methods of explicitly adding dissipation to central differencing schemes are discussed, as well as methods of achieving naturally dissipative schemes through upwind differencing. Also discussed are methods of capturing sharp discontinuities without introducing nonphysical oscillations into the computations.

2.3.1 Artificial Dissipation Models for Central-Difference Schemes

Because the basic numerical scheme uses central differences to represent spatial derivatives, the artificial dissipation required to avoid spurious oscillations in the vicinity of shocks and to stabilize the scheme is implemented in a convenient manner by modifying the convective fluxes:

$$\begin{aligned}
\hat{\mathbf{F}}_{i\pm\frac{1}{2},j,k} &= \frac{1}{2}\left(\hat{\mathbf{F}}_{i,j,k} + \hat{\mathbf{F}}_{i\pm1,j,k}\right) - \mathbf{d}_{i\pm\frac{1}{2},j,k} \\
\hat{\mathbf{G}}_{i,j\pm\frac{1}{2},k} &= \frac{1}{2}\left(\hat{\mathbf{G}}_{i,j,k} + \hat{\mathbf{G}}_{i,j\pm1,k}\right) - \mathbf{d}_{i,j\pm\frac{1}{2},k} \\
\hat{\mathbf{H}}_{i,j,k\pm\frac{1}{2}} &= \frac{1}{2}\left(\hat{\mathbf{H}}_{i,j,k} + \hat{\mathbf{H}}_{i,j,k\pm1}\right) - \mathbf{d}_{i,j,k\pm\frac{1}{2}}
\end{aligned} \tag{2.3.1}$$

The terms $\mathbf{d}_{i\pm\frac{1}{2},j,k}$, $\mathbf{d}_{i,j\pm\frac{1}{2},k}$, and $\mathbf{d}_{i,j,k\pm\frac{1}{2}}$ represent the dissipative terms in the i, j, and k directions, respectively. Although many variations in dissipation models are presented in the literature, only two specific forms are discussed in this paper.

Scalar Dissipation Model

The basic dissipation model is a nonisotropic model, where the dissipative terms are functions of the spectral radii of the Jacobian matrices associated with the appropriate coordinate directions. The details of the model vary with specific researchers, and no attempt is made here to describe the many variations. However, the essential ingredients are described below, and more details can be found in many excellent references (e.g., Refs. 1, 2, 3, 4, 5, 6 and 7)

For clarity, a detailed description of the dissipative terms for the i direction is given as

$$\begin{aligned}
\mathbf{d}_{i+\frac{1}{2},j,k} = \lambda_{i+\frac{1}{2},j,k}\big[&\epsilon^{(2)}_{i+\frac{1}{2},j,k}(\mathbf{W}_{i+1,j,k} - \mathbf{W}_{i,j,k}) \\
&- \epsilon^{(4)}_{i+\frac{1}{2},j,k}(\mathbf{W}_{i+2,j,k} - 3\mathbf{W}_{i+1,j,k} \\
&+ \quad 3\mathbf{W}_{i,j,k} - \mathbf{W}_{i-1,j,k})\big]
\end{aligned} \tag{2.3.2}$$

In the above expression, the coefficients $\epsilon^{(2)}$ and $\epsilon^{(4)}$ are related to the pressure gradient parameter ν.

$$\nu_i = \frac{|p_{i+1,j,k} - 2p_{i,j,k} + p_{i-1,j,k}|}{p_{i+1,j,k} + 2p_{i,j,k} + p_{i-1,j,k}} \tag{2.3.3}$$

$$\epsilon^{(2)}_{i+\frac{1}{2},j,k} = \kappa^{(2)}\max(\nu_{i+1}, \nu_i) \tag{2.3.4}$$

$$\epsilon^{(4)}_{i+\frac{1}{2},j,k} = \max\left\{0, \left(\kappa^{(4)} - \epsilon^{(2)}_{i+\frac{1}{2},j,k}\right)\right\} \tag{2.3.5}$$

where $\kappa^{(2)}$ and $\kappa^{(4)}$ are constants with typical values of 1/2 and 1/64, respectively. The variable $\mathbf{W}$ is related to the solution vector $\mathbf{w}$ by the equation

$$\mathbf{W} = \mathbf{w} + [0, 0, 0, 0, p]^T \tag{2.3.6}$$

The term $\lambda_{i+\frac{1}{2},j,k}$ is the scaling factor associated with the ξ–coordinate. This scale factor is defined as

$$\lambda_{i+\frac{1}{2},j,k} = \frac{1}{2}\left[(\bar{\lambda}_\xi)_{i,j,k} + (\lambda_\xi)_{i+1,j,k}\right] \tag{2.3.7}$$

where $\bar{\lambda}_\xi$ is related to the spectral radii of the flux Jacobian matrix in the three coordinate directions as follows

$$(\bar{\lambda}_\xi) = (\lambda_\xi)\left[1 + \left(\frac{\lambda_\eta}{\lambda_\xi}\right)^{0.5} + \left(\frac{\lambda_\zeta}{\lambda_\xi}\right)^{0.5}\right] \tag{2.3.8}$$

The spectral radii for the ξ, η, and ζ directions are

$$\begin{aligned}
\lambda_\xi &= |q_\xi| + a\sqrt{\xi_x^2 + \xi_y^2 + \xi_z^2} \\
\lambda_\eta &= |q_\eta| + a\sqrt{\eta_x^2 + \eta_y^2 + \eta_z^2} \\
\lambda_\zeta &= |q_\zeta| + a\sqrt{\zeta_x^2 + \zeta_y^2 + \zeta_z^2}
\end{aligned} \tag{2.3.9}$$

In the above equations, a is the local speed of sound, and q_ξ, q_η, and q_ζ are the contravariant velocity vectors in the i, j, and k (ξ,η,ζ) directions, respectively, and are given by the relations

$$\begin{aligned}
q_\xi &= \xi_x u + \xi_y v + \xi_z w \\
q_\eta &= \eta_x u + \eta_y v + \eta_z w \\
q_\zeta &= \zeta_x u + \zeta_y v + \zeta_z w
\end{aligned} \tag{2.3.10}$$

Expressions for the artificial dissipation coefficients in the j and k directions can be derived in a similar manner and take the form

$$(\bar{\lambda}_\eta) = (\lambda_\eta)\left[1 + \left(\frac{\lambda_\xi}{\lambda_\eta}\right)^{0.5} + \left(\frac{\lambda_\zeta}{\lambda_\eta}\right)^{0.5}\right] \tag{2.3.11}$$

$$(\bar{\lambda}_\zeta) = (\lambda_\zeta)\left[1 + \left(\frac{\lambda_\xi}{\lambda_\zeta}\right)^{0.5} + \left(\frac{\lambda_\eta}{\lambda_\zeta}\right)^{0.5}\right] \tag{2.3.12}$$

The pressure switch given in Eqs. (2.3.3), (2.3.4), and (2.3.5) serves to increase the second difference dissipation while switching off the fourth differences near discontinuities. This formulation is given by Jameson et al.[5] and increases the sharpness of the shock wave without the introduction of unwanted oscillations due to differencing across the shock.

Matrix-Valued Dissipation

The dissipation model described above is not optimal in the sense that the same dissipation scaling is used for all the governing equations in a given coordinate direction. Reduced artificial dissipation can be obtained by individually scaling the dissipation contribution to each equation, as is done implicitly in upwind schemes.[8,9] The scalar coefficients used in the artificial dissipation model are replaced by the modulus (absolute values) of flux Jacobian matrices. Thus Eqs. (2.3.8)-(2.3.12) can be rewritten as

$$\lambda_\xi = |\mathbf{A}|,\ \lambda_\eta = |\mathbf{B}|,\ \lambda_\zeta = |\mathbf{C}| \tag{2.3.13}$$

The matrices **A**, **B**, and **C** have very few nonzero elements and can be found in Ref. 7 in their entirety. The absolute value of these matrices, illustrated here for the matrix **A**, is defined in the following manner. Let

$$\mathbf{A} = \mathbf{T}_\xi \mathbf{\Lambda}_\xi \mathbf{T}_\xi^{-1} \tag{2.3.14}$$

where $\mathbf{\Lambda}_\xi$ is a diagonal matrix with the eigenvalues of **A** as its elements. Then

$$|\mathbf{A}| = \mathbf{T}_\xi |\mathbf{\Lambda}_\xi| \mathbf{T}_\xi^{-1} \tag{2.3.15}$$

where

$$|\mathbf{\Lambda}_\xi| = \begin{bmatrix} |\lambda_1| & 0 & 0 & 0 & 0 \\ 0 & |\lambda_2| & 0 & 0 & 0 \\ 0 & 0 & |\lambda_3| & 0 & 0 \\ 0 & 0 & 0 & |\lambda_3| & 0 \\ 0 & 0 & 0 & 0 & |\lambda_3| \end{bmatrix} \tag{2.3.16}$$

The diagonal elements of the above matrix are

$$\begin{aligned} \lambda_1 &= q_\xi + a\sqrt{\xi_x^2 + \xi_y^2 + \xi_z^2} \\ \lambda_2 &= q_\xi - a\sqrt{\xi_x^2 + \xi_y^2 + \xi_z^2} \\ \lambda_3 &= q_\xi \end{aligned} \tag{2.3.17}$$

After considerable algebra $|\mathbf{A}|$ can be expressed in the following manner as

$$\begin{aligned} |\mathbf{A}| = {} & |\lambda_3|\,\mathbf{I} + \left(\frac{|\lambda_1| + |\lambda_2|}{2} - |\lambda_3|\right) \\ & \left[\frac{(\gamma - 1)}{a^2}\mathbf{E}_1 + \frac{\mathbf{E}_2}{\xi_x^2 + \xi_y^2 + \xi_z^2}\right] \\ & + \frac{|\lambda_1| - |\lambda_2|}{2a\sqrt{\xi_x^2 + \xi_y^2 + \xi_z^2}}(\mathbf{E}_3 + (\gamma - 1)\mathbf{E}_4) \end{aligned} \tag{2.3.18}$$

where **I** is the identity matrix, and $\mathbf{E}_1$, $\mathbf{E}_2$, $\mathbf{E}_3$, and $\mathbf{E}_4$ are given by the relations

$$\mathbf{E}_1 = \begin{bmatrix} V_t^2/2 & -u & -v & -w & 1 \\ uV_t^2/2 & -uu & -uv & -uw & u \\ vV_t^2/2 & -vu & -vv & -vw & v \\ wV_t^2/2 & -wu & -wv & -ww & w \\ hV_t^2/2 & -hu & -hv & -hw & h \end{bmatrix} \tag{2.3.19}$$

$$\mathbf{E}_2 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ -\xi_x q_\xi & \xi_x\xi_x & \xi_x\xi_y & \xi_x\xi_z & 0 \\ -\xi_y q_\xi & \xi_y\xi_x & \xi_y\xi_y & \xi_y\xi_z & 0 \\ -\xi_z q_\xi & \xi_z\xi_x & \xi_z\xi_y & \xi_z\xi_z & 0 \\ -q_\xi q_\xi & q_\xi\xi_x & q_\xi\xi_y & q_\xi\xi_z & 0 \end{bmatrix} \tag{2.3.20}$$

$$\mathbf{E}_3 = \begin{bmatrix} -q_\xi & \xi_x & \xi_y & \xi_z & 0 \\ -uq_\xi & u\xi_x & u\xi_y & u\xi_z & 0 \\ -vq_\xi & v\xi_x & v\xi_y & v\xi_z & 0 \\ -wq_\xi & w\xi_x & w\xi_y & w\xi_z & 0 \\ -hq_\xi & h\xi_x & h\xi_y & h\xi_z & 0 \end{bmatrix} \tag{2.3.21}$$

$$\mathbf{E}_4 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ \xi_x V_t^2/2 & -\xi_x u & -\xi_x v & -\xi_x w & \xi_x \\ \xi_y V_t^2/2 & -\xi_y u & -\xi_y v & -\xi_y w & \xi_y \\ \xi_z V_t^2/2 & -\xi_z u & -\xi_z v & -\xi_z w & \xi_z \\ q_\xi V_t^2/2 & -q_\xi u & -q_\xi v & -q_\xi w & q_\xi \end{bmatrix} \tag{2.3.22}$$

and $V_t^2 = u^2 + v^2 + w^2$ is the total velocity.

By taking advantage of the special form of the elements of $|\mathbf{A}|$, one can evaluate the matrix-vector products of the form $|\mathbf{A}|(W_{i+1,j,k} - W_{i,j,k})$ very efficiently, without ever evaluating $|\mathbf{A}|$ directly.

In practice, λ_1, λ_2, λ_3 cannot be chosen as given by Eq. (2.3.17). Near stagnation points, λ_3 approaches zero; near sonic lines, λ_1 or λ_2 approaches zero. Since zero artificial viscosity can create numerical difficulties, these values are limited in the following manner

$$\begin{aligned} |\tilde{\lambda}_1| &= \max(|\lambda_1|, V_n\lambda_\xi) \\ |\tilde{\lambda}_2| &= \max(|\lambda_2|, V_n\lambda_\xi) \\ |\tilde{\lambda}_3| &= \max(|\lambda_3|, V_l\lambda_\xi) \end{aligned} \tag{2.3.23}$$

where V_n limits the eigenvalues associated with the nonlinear characteristic fields to a minimum value that is a fraction of the spectral radius, while V_l provides a similar limiter for the eigenvalues associated with the linear characteristic fields. The values for the limiting coefficients V_n and V_l are determined through numerical experimentation such that sharper shocks and suction peaks are captured without the introduction of spurious oscillations in the solution while still maintaining good convergence properties. By setting $V_n = V_l = 1$, we recover the scalar form of the artificial dissipation, whereas $V_n = V_l = 0$ corresponds to the use of actual eigenvalues without any limiters. The eigenvalues obtained from Eq. (2.3.23) are then modified for large changes in cell aspect ratios with the expressions given in Eq. (2.3.8). Similar expressions can be derived for the matrices **B** and **C** by replacing the contravariant velocity q_ξ by q_η and q_ζ, and ξ by η and ζ, respectively, in Eqs. (2.3.14) through (2.3.22).

Note that if the dissipative fluxes are interpreted as modifiers to the physical fluxes at the interfaces of the difference molecule, then the central-difference scheme with the matrix dissipation closely resembles the characteristic decomposition used in upwind schemes.[10,11]

A comparison of results obtained with both scalar and matrix dissipation is shown in Fig. 2.3.1. These results have been obtained with the code described in Ref. 9. The case shown is the inviscid flow over an NACA 0012 airfoil at a free-stream Mach number of 0.8 and an angle of attack of 1.25°. This flow field is characterized by a moderate strength shock on the upper surface and a much weaker shock on the lower surface. As seen in the figure, the results obtained with both the scalar and matrix dissipation are very similar. However,

the solution obtained with matrix dissipation exhibits slightly sharper shocks on both the upper and lower surfaces.

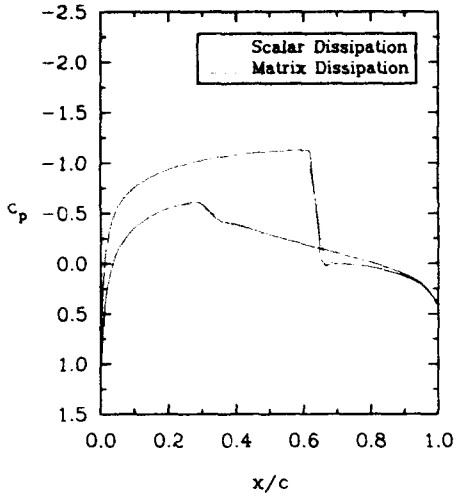


Figure 2.3.1 Central-difference scheme results with scalar and matrix dissipation for NACA 0012 with $M_\infty = 0.8$ and $\alpha = 1.25°$.

2.3.2 Upwind Differencing

Flux-Vector Splitting

For flux-vector splitting, the fluxes in generalized coordinates, $\hat{\mathbf{F}}$, $\hat{\mathbf{G}}$, and $\hat{\mathbf{H}}$ are split into forward and backward contributions according to the signs of the eigenvalues of the Jacobian matrices and are differenced accordingly. For example, the flux in the ξ direction can be differenced as

$$\delta_\xi \hat{\mathbf{F}} = \delta_\xi^- \hat{\mathbf{F}}^+ + \delta_\xi^+ \hat{\mathbf{F}}^- \tag{2.3.24}$$

because $\hat{\mathbf{F}}^+$ has all nonnegative eigenvalues and $\hat{\mathbf{F}}^-$ has all nonpositive eigenvalues. Two methods of flux-vector splitting are discussed below.

The first method is the technique outlined by Steger and Warming in Ref. 12. Since the flux vectors are homogeneous functions of degree one in $\mathbf{w}$, they can be expressed in terms of their Jacobian matrices. For example, considering the flux vector in the ξ direction, $\hat{\mathbf{F}}$ can then be written as

$$\hat{\mathbf{F}} = \hat{\mathbf{A}}\hat{\mathbf{w}} = \frac{\partial \hat{\mathbf{F}}}{\partial \hat{\mathbf{w}}}\hat{\mathbf{w}} \tag{2.3.25}$$

A similarity transformation allows Eq. (2.3.25) to be written as

$$\hat{\mathbf{F}} = \hat{\mathbf{A}}\hat{\mathbf{w}} = \mathbf{T}_\xi \mathbf{\Lambda}_\xi \mathbf{T}_\xi^{-1}\hat{\mathbf{w}} \tag{2.3.26}$$

where the matrix $\mathbf{\Lambda}_\xi$ is a diagonal matrix composed of the eigenvalues of $\hat{\mathbf{A}}$ and is given by

$$\mathbf{\Lambda}_\xi = \begin{bmatrix} \lambda_1 & 0 & 0 & 0 & 0 \\ 0 & \lambda_2 & 0 & 0 & 0 \\ 0 & 0 & \lambda_3 & 0 & 0 \\ 0 & 0 & 0 & \lambda_4 & 0 \\ 0 & 0 & 0 & 0 & \lambda_5 \end{bmatrix} \tag{2.3.27}$$

where

$$\begin{aligned} \lambda_{1,2,3} &= q_\xi = \xi_x u + \xi_y v + \xi_z w \\ \lambda_4 &= q_\xi + a\sqrt{\xi_x^2 + \xi_y^2 + \xi_z^2} \\ \lambda_5 &= q_\xi - a\sqrt{\xi_x^2 + \xi_y^2 + \xi_z^2} \end{aligned} \tag{2.3.28}$$

The eigenvalues can then be decomposed into nonnegative and nonpositive components

$$\lambda_i = \lambda_i^+ + \lambda_i^- \tag{2.3.29}$$

where

$$\lambda_i^\pm = \frac{\lambda_i \pm |\lambda_i|}{2} \tag{2.3.30}$$

Similarly, the eigenvalue matrix $\mathbf{\Lambda}_\xi$ can be decomposed into

$$\mathbf{\Lambda}_\xi = \mathbf{\Lambda}_\xi^+ + \mathbf{\Lambda}_\xi^- \tag{2.3.31}$$

where $\mathbf{\Lambda}_\xi^+$ is made up of the nonnegative contributions of λ_i^+ and $\mathbf{\Lambda}_\xi^-$ is constructed of the nonpositive contributions of λ_i^-. This splitting of the eigenvalue matrix, combined with Eq. (2.3.26), allows the flux vector $\hat{\mathbf{F}}$ to be rewritten as

$$\begin{aligned} \hat{\mathbf{F}} &= \mathbf{T}_\xi(\mathbf{\Lambda}_\xi^+ + \mathbf{\Lambda}_\xi^-)\mathbf{T}_\xi^{-1}\hat{\mathbf{w}} \\ &= (\mathbf{A}^+ + \mathbf{A}^-)\hat{\mathbf{w}} = \hat{\mathbf{F}}^+ + \hat{\mathbf{F}}^- \end{aligned} \tag{2.3.32}$$

The flux vector $\hat{\mathbf{F}}$ has three distinct eigenvalues given by Eq. (2.3.28) and can therefore be written as a sum of three subvectors, each of which has a distinct eigenvalue as a coefficient[13]

$$\hat{\mathbf{F}} = \hat{\mathbf{F}}_1 + \hat{\mathbf{F}}_2 + \hat{\mathbf{F}}_3 \tag{2.3.33}$$

where

$$\hat{\mathbf{F}}_1 = \lambda_1 \frac{\gamma - 1}{J\gamma} \begin{Bmatrix} \rho \\ \rho u \\ \rho v \\ \rho w \\ \frac{\rho}{2}(u^2 + v^2 + w^2) \end{Bmatrix} \tag{2.3.34}$$

$$\hat{\mathbf{F}}_{2,3} = \lambda_{4,5} \frac{1}{J2\gamma} \begin{Bmatrix} \rho \\ \rho u \pm pa\hat{\xi}_x \\ \rho v \pm pa\hat{\xi}_y \\ \rho w \pm pa\hat{\xi}_z \\ e + p \pm \frac{pa}{\sqrt{\xi_x^2 + \xi_y^2 + \xi_z^2}} \end{Bmatrix} \tag{2.3.35}$$

and the direction cosines of the directed interface in the ξ direction are

$$\begin{aligned} \hat{\xi}_x &= \frac{\xi_x}{|\nabla \xi|} \\ \hat{\xi}_y &= \frac{\xi_y}{|\nabla \xi|} \\ \hat{\xi}_z &= \frac{\xi_z}{|\nabla \xi|} \end{aligned} \tag{2.3.36}$$

where

$$|\nabla \xi| = \sqrt{\xi_x^2 + \xi_y^2 + \xi_z^2} \tag{2.3.37}$$

The forward and backward flux vectors $\hat{\mathbf{F}}^+$ and $\hat{\mathbf{F}}^-$ are formed from Eq. (2.3.33), (2.3.34), and (2.3.35) by inserting $\lambda_i = \lambda_i^+$ and $\lambda_i = \lambda_i^-$, respectively.

For supersonic and sonic flow in the ξ direction (i.e., $|M_\xi| = |\bar{u}/a| \geq 1$), where $\bar{u} = q_\xi/|\nabla\xi|$ represents the velocity normal to a ξ = Constant face, the fluxes in this direction become

$$\begin{aligned} \hat{\mathbf{F}}^+ &= \hat{\mathbf{F}} \quad \hat{\mathbf{F}}^- = 0 \qquad (M_\xi \geq 1) \\ \hat{\mathbf{F}}^+ &= 0 \quad \hat{\mathbf{F}}^- = \hat{\mathbf{F}} \qquad (M_\xi \leq -1) \end{aligned} \tag{2.3.38}$$

The split fluxes in the other two directions are easily obtained by interchanging η or ζ in place of ξ.

The fluxes split in the aforementioned manner are not continuously differentiable at zeros of the eigenvalues (i.e., sonic and stagnation points). (See Ref. 14.) This is illustrated in Fig. 2.3.2, where the split mass flux contributions for the one-dimensional Euler equations, nondimensionalized by ρa, are shown as a function of Mach number. The gradient discontinuities in the split fluxes are evident as the eigenvalues pass through zero. The lack of differentiability of the split fluxes has been shown in some cases to cause small oscillations at sonic points that are rarely noticeable for most aerodynamic applications but can be remedied by biasing the eigenvalues near zero to a small value.[14]

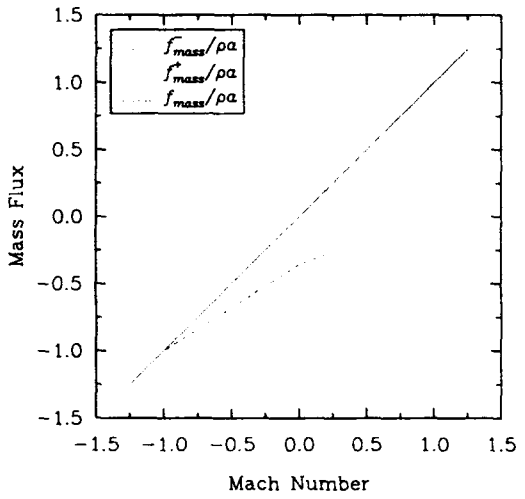


Figure 2.3.2 Variation of Steger and Warming split mass flux with Mach number.

The Jacobian matrices of $\hat{\mathbf{F}}^+$ and $\hat{\mathbf{F}}^-$ that are required for proper linearization for an implicit scheme do not share the same eigenvalues as $\mathbf{A}^+$ and $\mathbf{A}^-$ defined in Eq. (2.3.32). However, the Jacobian matrices of $\hat{\mathbf{F}}^+$ and $\hat{\mathbf{F}}^-$ do have eigenvalues with the same signs as $\mathbf{A}^+$ and $\mathbf{A}^-$ so that upwind differencing of the spatial derivatives remains appropriate.[14] Although $\mathbf{A}^\pm$ are easier to form, their use in implicit schemes, instead of the correct linearizations ($\partial\hat{\mathbf{F}}^\pm/\partial\hat{\mathbf{w}}$), has been shown in many cases to cause severe time-step limitations.[14,15,16]

In 1982, a new method of splitting the flux vector was proposed by Van Leer.[17] Here, the fluxes are split so that the forward and backward flux contributions blend smoothly at eigenvalue sign changes (i.e., near sonic and stagnation points). Just as for the Steger-Warming splitting, the Jacobian matrices $\partial\hat{\mathbf{F}}^+/\partial\hat{\mathbf{w}}$ must have nonnegative eigenvalues and $\partial\hat{\mathbf{F}}^-/\partial\hat{\mathbf{w}}$ must have nonpositive eigenvalues so that upwind differencing can be used for the spatial derivatives. In addition, both Jacobians have one zero eigenvalue for subsonic Mach numbers, which leads to steady transonic shock structures with only two transition zones.[17] In practice, when second-order spatial differencing is used, shocks with only one interior zone are usually obtained.[18] This feature is not observed with the Steger-Warming flux splitting.

The three-dimensional splittings of Van Leer are given below for the ξ direction; the others are obtained similarly. The flux is split according to the contravariant Mach number in the ξ direction, which is defined previously as $|M_\xi| = |\bar{u}/a| \geq 1$. For supersonic flow ($|M_\xi| \geq 1$),

$$\begin{aligned} \hat{\mathbf{F}}^+ &= \hat{\mathbf{F}} \quad \hat{\mathbf{F}}^- = 0 \qquad (M_\xi \geq 1) \\ \hat{\mathbf{F}}^+ &= 0 \quad \hat{\mathbf{F}}^- = \hat{\mathbf{F}} \qquad (M_\xi \leq -1) \end{aligned} \tag{2.3.39}$$

and for subsonic flow ($|M_\xi| \leq 1$),

$$\hat{\mathbf{F}}^\pm = \frac{|\nabla\xi|}{J} \begin{Bmatrix} f^\pm_{\text{mass}} \\ f^\pm_{\text{mass}}\left\{\left[\hat{\xi}_x(-\bar{u} \pm 2a)/\gamma\right] + u\right\} \\ f^\pm_{\text{mass}}\left\{\left[\hat{\xi}_y(-\bar{u} \pm 2a)/\gamma\right] + v\right\} \\ f^\pm_{\text{mass}}\left\{\left[\hat{\xi}_z(-\bar{u} \pm 2a)/\gamma\right] + w\right\} \\ f^\pm_{\text{energy}} \end{Bmatrix} \tag{2.3.40}$$

where

$$f^\pm_{\text{mass}} = \pm\rho a(M_\xi \pm 1)^2/4 \tag{2.3.41}$$

$$\begin{aligned} f^\pm_{\text{energy}} = f^\pm_{\text{mass}}\{&\left[-(\gamma-1)\bar{u}^2 \pm 2(\gamma-1)\bar{u}a + 2a^2\right]/(\gamma^2-1) \\ &+ (u^2+v^2+w^2)/2\} \end{aligned} \tag{2.3.42}$$

To form $\hat{\mathbf{F}}^\pm$, the direction cosines $\hat{\xi}_x$, $\hat{\xi}_y$, and $\hat{\xi}_z$ are given by Eq. (2.3.36) and $\bar{u}$ is the velocity normal to a ξ = Constant face. The fluxes in the other two directions are easily formed by interchanging ξ with η or ζ. In Fig. 2.3.3 the nondimensionalized mass flux using the Van Leer splitting is shown as a function of Mach number for the one-dimensional Euler equations. The split fluxes are continuously differentiable at sonic and stagnation points; the improvement over the Steger-Warming splitting is apparent.

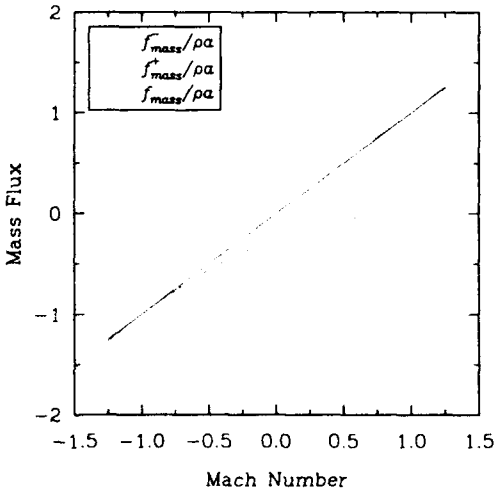


Figure 2.3.3 Variation of Van Leer split mass flux with Mach number.

An interesting numerical consequence of splitting the fluxes can be observed by examining the differencing of the mass and energy fluxes. Considering a simplified one-dimensional case, an approximation to the spatial derivatives of the mass and energy flux is given by

$$\text{mass} \quad \frac{\partial(\rho u)}{\partial x} \approx \frac{(\rho u)_{i+\frac{1}{2}} - (\rho u)_{i-\frac{1}{2}}}{\Delta x}$$
$$\text{energy} \quad \frac{\partial(\rho u H)}{\partial x} \approx \frac{(\rho u H)_{i+\frac{1}{2}} - (\rho u H)_{i-\frac{1}{2}}}{\Delta x} \qquad (2.3.43)$$

At a steady state, $(\rho u)_{i+1/2} = (\rho u)_{i-1/2}$ so the energy equation can be written as

$$\frac{\partial(\rho u H)}{\partial x} \approx (\rho u)_{i\pm\frac{1}{2}} \frac{H_{i+\frac{1}{2}} - H_{i-\frac{1}{2}}}{\Delta x} \qquad (2.3.44)$$

Because this will also equal zero at a steady state, the total enthalpy will be preserved numerically as it should be physically. When the fluxes are split with either the Steger-Warming or Van Leer method (as well as flux-difference splitting), it is no longer possible to obtain a similar relation because the mass and energy fluxes are differenced in different manners. The consequence is that, although mass, momentum, and energy are conserved by the numerical algorithm, the total enthalpy is not conserved. In 1987, Hänel[19] modified the energy formulation for the Van Leer flux-vector splitting so that the split energy flux is given simply by

$$f^{\pm}_{\text{energy}} = f^{\pm}_{\text{mass}} H \qquad (2.3.45)$$

By defining the energy flux in this way, the total enthalpy is preserved. In addition, because the total enthalpy is a constant at steady state, the energy flux remains degenerate so that the shock-capturing capabilities are not compromised.

Godunov's Method

A very successful scheme for solving the Euler equations that has led to significant improvements in the accuracy of modern numerical algorithms is due to Godunov (e.g., see Ref. 20). For this scheme, a piecewise constant approximation of the data in each cell is obtained, which represents the average of the data in the cell

$$\mathbf{w}_i = \frac{1}{\Delta x} \int_{i-\frac{1}{2}}^{i+\frac{1}{2}} \mathbf{w}(x, t) dx \qquad (2.3.46)$$

Each cell interface, located at $i \pm 1/2$, is then considered to separate two regions of constant properties in the same manner as a diaphragm separates regions of high- and low-pressure gas in a shock tube. Because an exact solution exists for this problem, the evolution of the flow field can be easily determined by solving for the interaction of the resulting wave system, provided that waves from neighboring cells do not interact. Afterwards, the solution in each cell is averaged, and the process is repeated. The process described above is summarized in Figs. 2.3.4, 2.3.5, and 2.3.6.

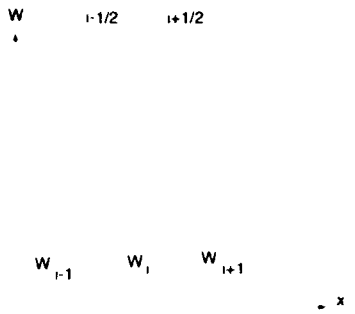


Figure 2.3.4 Average state at time t = n.

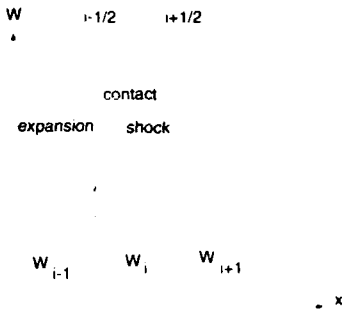


Figure 2.3.5 Solution of local Riemann problem.

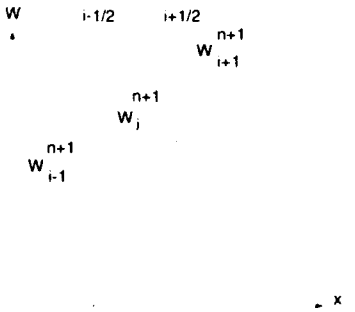


Figure 2.3.6 Average state at time t = n + 1.

To advance the solution in time, the one-dimensional time-dependent Euler equations are integrated over both space and time to yield an equation that describes the time evolution of the cell average in each cell. For example, in cell i,

$$\mathbf{w}_i^{n+1} = \mathbf{w}_i^n - \frac{\Delta t}{\Delta x}\left[\mathbf{F}\left(\mathbf{w}_{i+\frac{1}{2}}\right) - \mathbf{F}\left(\mathbf{w}_{i-\frac{1}{2}}\right)\right] \qquad (2.3.47)$$

Here, $\mathbf{F}\left(\mathbf{w}_{i\pm\frac{1}{2}}\right)$ represents the time average of the flux between times n and n + 1. Recall that in advancing the solution in time, Δt is chosen so that there is no interaction of the waves from neighboring cell faces. Therefore, the solution at the interface is constant over the time interval of interest. The solution can then be advanced in time by forming the fluxes

on the faces from the data obtained by solving the Riemann problem and advancing the solution using Eq. (2.3.47).

The above process can be broken down into a "projection" and an "evolution" stage as described by Van Leer.[21] In the projection stage, the behavior of the data in each cell is reconstructed whereas the evolution stage refers to the solution of the Riemann problem. In Godunov's method, the data in the cell are reconstructed by assuming it to be piecewise constant, which leads to first-order spatial accuracy. By replacing this approximation by a piecewise linear representation of the data, the accuracy of the scheme can be raised to second order.[22]

Osher's Scheme

In Godunov's technique, the solution of the Riemann problem requires an iterative procedure at each interface whenever a shock wave is present. To circumvent the iterative process, an approximate solution to the Riemann problem can be obtained by replacing a shock wave with an overturned rarefaction.[23,24] Therefore, because all nonlinear waves are expressed in terms of rarefactions, explicit relations are obtained for the intermediate state variables connected by each wave.

Flux-Difference Splitting

For flux-difference splitting, the solution of the Riemann problem is again considered. However, the solution of the Riemann problem is foregone in favor of an exact solution to an approximate problem that does not require any iteration. More specifically, for one space dimension, data are advanced in time through a linearized version of the Euler equations given by

$$\frac{\partial \mathbf{w}}{\partial t} + \tilde{\mathbf{A}}\frac{\partial \mathbf{w}}{\partial x} = 0 \tag{2.3.48}$$

where $\tilde{\mathbf{A}}$ is a specially constructed constant matrix that satisfies the property that for any $\mathbf{w_L}$ and $\mathbf{w_R}$ (which represent the left and right state variables on either side of a cell face),

$$\tilde{\mathbf{A}}(\mathbf{w_L}, \mathbf{w_R})\Delta\mathbf{w} = \Delta\mathbf{F} \tag{2.3.49}$$

where $\Delta(\cdot) = (\cdot)_L - (\cdot)_R$ (i.e. the jump across an interface). Note that the tilde ($\sim$) denotes that the matrix is constructed with a specific averaging procedure that is described below.

Equation (2.3.49) can also be written as

$$\tilde{\mathbf{T}}\tilde{\mathbf{\Lambda}}\tilde{\mathbf{T}}^{-1}\Delta\mathbf{w} = \Delta\mathbf{F} \tag{2.3.50}$$

Since the eigenvalues represent wave speeds of individual waves, $\Delta\mathbf{q} = \tilde{\mathbf{T}}^{-1}\Delta\mathbf{w}$ represent jumps in the variables due to the influence of each wave. Hence, the change in flux between the left and right states is expressed in terms of the jumps in these states projected onto the eigenvectors:

$$\Delta\mathbf{F} = \sum(\tilde{\lambda}_k\Delta\mathbf{q})\tilde{\mathbf{T}}_k \tag{2.3.51}$$

By considering the backward-moving ($\tilde{\lambda}_k < 0$) and forward-moving waves ($\tilde{\lambda}_k > 0$) separately, the flux on the face at $i+\frac{1}{2}$ in Fig. 2.3.5 can be determined through any of the following equations (all of which are equivalent):

$$\mathbf{F}_{i+\frac{1}{2}} = \mathbf{F_L} + \sum_{\tilde{\lambda}<0}(\tilde{\lambda}_k\Delta\mathbf{q})\tilde{\mathbf{T}}_k \tag{2.3.52}$$

$$\mathbf{F}_{i+\frac{1}{2}} = \mathbf{F_R} - \sum_{\tilde{\lambda}>0}(\tilde{\lambda}_k\Delta\mathbf{q})\tilde{\mathbf{T}}_k \tag{2.3.53}$$

$$\mathbf{F}_{i+\frac{1}{2}} = \frac{1}{2}(\mathbf{F_L} + \mathbf{F_R}) - \frac{1}{2}\sum_k(|\tilde{\lambda}_k|\Delta\mathbf{q})\tilde{\mathbf{T}}_k \tag{2.3.54}$$

The last form can be considered to represent a central-difference term plus a dissipative term.

Note that if $\mathbf{w_L}$ and $\mathbf{w_R}$ are such that only a single discontinuity is present, Eq. (2.3.49) reduces to

$$S\Delta\mathbf{w} = \Delta\mathbf{F} \tag{2.3.55}$$

where S is the speed of the discontinuity and is an eigenvalue of $\tilde{\mathbf{A}}$. In this case, the discontinuity will be resolved exactly.

The specific determination of $\tilde{\mathbf{A}}$ is presented in Ref. 25. Here, note that the flux $\mathbf{F}$ and dependent variables $\mathbf{w}$ are both quadratic in the components of a parameter vector given by

$$\mathbf{w}' = \sqrt{\rho}(1, u, v, w, H)^T \tag{2.3.56}$$

With this relation, the jumps across a wave in both the dependent variables and the fluxes can be expressed in terms of jumps in the parameter vector

$$\Delta\mathbf{w} = \tilde{\mathbf{B}}\Delta\mathbf{w}' \tag{2.3.57}$$

$$\Delta\mathbf{F} = \tilde{\mathbf{C}}\Delta\mathbf{w}' \tag{2.3.58}$$

In Ref. 25, the exact form of $\tilde{\mathbf{B}}$ and $\tilde{\mathbf{C}}$ are shown to be reasonably simple. For example, for three-dimensional Cartesian coordinates,

$$\tilde{\mathbf{B}} = \begin{bmatrix} 2\bar{w}_1 & 0 & 0 & 0 & 0 \\ \bar{w}_2 & \bar{w}_1 & 0 & 0 & 0 \\ \bar{w}_3 & 0 & \bar{w}_1 & 0 & 0 \\ \bar{w}_4 & 0 & 0 & \bar{w}_1 & 0 \\ \frac{\bar{w}_5}{\gamma} & \frac{\gamma-1}{\gamma}\bar{w}_2 & \frac{\gamma-1}{\gamma}\bar{w}_3 & \frac{\gamma-1}{\gamma}\bar{w}_4 & \frac{\bar{w}_1}{\gamma} \end{bmatrix} \tag{2.3.59}$$

$$\tilde{\mathbf{C}} = \begin{bmatrix} \bar{w}_2 & \bar{w}_1 & 0 & 0 & 0 \\ \frac{\gamma-1}{\gamma}\bar{w}_5 & \frac{\gamma-1}{\gamma}\bar{w}_2 & \frac{\gamma-1}{\gamma}\bar{w}_3 & \frac{\gamma-1}{\gamma}\bar{w}_4 & \frac{\gamma-1}{\gamma}\bar{w}_1 \\ 0 & \bar{w}_3 & \bar{w}_2 & 0 & 0 \\ 0 & \bar{w}_4 & 0 & \bar{w}_2 & 0 \\ 0 & \bar{w}_5 & 0 & 0 & \bar{w}_2 \end{bmatrix} \tag{2.3.60}$$

Here, the overbar denotes the arithmetic average of $\mathbf{w}'$ from the left and right states. From Eqs. (2.3.57) and (2.3.58), the matrix $\tilde{\mathbf{A}}$ is given by

$$\tilde{\mathbf{A}} = \tilde{\mathbf{C}}\tilde{\mathbf{B}}^{-1} \tag{2.3.61}$$

This matrix corresponds to the Jacobian matrix $\mathbf{A} = \partial\mathbf{F}/\partial\mathbf{w}$, evaluated at an averaged state given by

$$\begin{aligned} \tilde{\rho} &= (\rho_L\rho_R)^{1/2} \\ \tilde{u} &= \frac{\rho_L^{1/2}u_L + \rho_R^{1/2}u_R}{\rho_L^{1/2} + \rho_R^{1/2}} \\ \tilde{v} &= \frac{\rho_L^{1/2}v_L + \rho_R^{1/2}v_R}{\rho_L^{1/2} + \rho_R^{1/2}} \\ \tilde{w} &= \frac{\rho_L^{1/2}w_L + \rho_R^{1/2}w_R}{\rho_L^{1/2} + \rho_R^{1/2}} \\ \tilde{H} &= \frac{\rho_L^{1/2}H_L + \rho_R^{1/2}H_R}{\rho_L^{1/2} + \rho_R^{1/2}} \end{aligned} \tag{2.3.62}$$

For three-dimensional flows in generalized coordinates, the flux on each face is determined in a one-dimensional manner, based on the assumption that the waves travel in the directions of the grid lines. Considering the ξ direction as an example, the flux is determined on a cell face located at $i+\frac{1}{2}$ with Eq. (2.3.54), where $\Delta\mathbf{q} = \tilde{\mathbf{T}}_\xi^{-1}\Delta\mathbf{w}$, $\tilde{\mathbf{T}} = \tilde{\mathbf{T}}_\xi$, and the subscript ξ denotes that the matrices are formed from the vectors normal to a ξ = Constant face. The matrices, $\tilde{\mathbf{T}}_\xi$ and $\tilde{\mathbf{T}}_\xi^{-1}$ are formed with the averaged values from Eq. (2.3.62) as are the eigenvalues. The resulting matrices are given as

$$\mathbf{T}_\xi = \begin{bmatrix} 1 & \hat{\xi}_x & \hat{\xi}_y & \hat{\xi}_z & 1 \\ u - a\hat{\xi}_x & \left(u - a\hat{\xi}_x\right)\hat{\xi}_x + a & \left(u - a\hat{\xi}_x\right)\hat{\xi}_y & \left(u - a\hat{\xi}_x\right)\hat{\xi}_z & u + a\hat{\xi}_x \\ v - a\hat{\xi}_y & \left(v - a\hat{\xi}_y\right)\hat{\xi}_x & \left(v - a\hat{\xi}_y\right)\hat{\xi}_y + a & \left(v - a\hat{\xi}_y\right)\hat{\xi}_z & v + a\hat{\xi}_y \\ w - a\hat{\xi}_z & \left(w - a\hat{\xi}_z\right)\hat{\xi}_x & \left(w - a\hat{\xi}_z\right)\hat{\xi}_y & \left(w - a\hat{\xi}_z\right)\hat{\xi}_z + a & w + a\hat{\xi}_z \\ H - \bar{u}a & \left(\frac{q^2}{2} - \bar{u}a\right)\hat{\xi}_x + ua & \left(\frac{q^2}{2} - \bar{u}a\right)\hat{\xi}_y + va & \left(\frac{q^2}{2} - \bar{u}a\right)\hat{\xi}_z + wa & H + \bar{u}a \end{bmatrix} \tag{2.3.63}$$

$$\mathbf{T}_\xi^{-1} = \begin{bmatrix} \frac{1}{2}\left(\frac{u}{a} + \frac{\gamma-1}{2}M^2\right) & -\frac{\hat{\xi}_x}{2a} - \frac{\gamma-1}{2}\frac{u}{a} & -\frac{\hat{\xi}_y}{2a} - \frac{\gamma-1}{2}\frac{v}{a} & -\frac{\hat{\xi}_z}{2a} - \frac{\gamma-1}{2}\frac{w}{a} & \frac{\gamma-1}{2a^2} \\ \hat{\xi}_x\left(1 + \frac{u}{a} - \frac{\gamma-1}{2}M^2\right) - \frac{u}{a} & \frac{\gamma-1}{a}\frac{u}{a}\hat{\xi}_x + \frac{1-\hat{\xi}_x^2}{a} & \frac{\gamma-1}{a}\frac{v}{a}\hat{\xi}_x - \frac{\hat{\xi}_x\hat{\xi}_y}{a} & \frac{\gamma-1}{a}\frac{w}{a}\hat{\xi}_x - \frac{\hat{\xi}_x\hat{\xi}_z}{a} & -\frac{\gamma-1}{a^2}\hat{\xi}_x \\ \hat{\xi}_y\left(1 + \frac{u}{a} - \frac{\gamma-1}{2}M^2\right) - \frac{v}{a} & \frac{\gamma-1}{a}\frac{u}{a}\hat{\xi}_y - \frac{\hat{\xi}_x\hat{\xi}_y}{a} & \frac{\gamma-1}{a}\frac{v}{a}\hat{\xi}_y + \frac{1-\hat{\xi}_y^2}{a} & \frac{\gamma-1}{a}\frac{w}{a}\hat{\xi}_y - \frac{\hat{\xi}_y\hat{\xi}_z}{a} & -\frac{\gamma-1}{a^2}\hat{\xi}_y \\ \hat{\xi}_z\left(1 + \frac{u}{a} - \frac{\gamma-1}{2}M^2\right) - \frac{w}{a} & \frac{\gamma-1}{a}\frac{u}{a}\hat{\xi}_z - \frac{\hat{\xi}_x\hat{\xi}_z}{a} & \frac{\gamma-1}{a}\frac{v}{a}\hat{\xi}_z - \frac{\hat{\xi}_y\hat{\xi}_z}{a} & \frac{\gamma-1}{a}\frac{w}{a}\hat{\xi}_z + \frac{1-\hat{\xi}_z^2}{a} & -\frac{\gamma-1}{a^2}\hat{\xi}_z \\ \frac{1}{2}\left(-\frac{u}{a} + \frac{\gamma-1}{2}M^2\right) & \frac{\hat{\xi}_x}{2a} - \frac{\gamma-1}{2a}\frac{u}{a} & \frac{\hat{\xi}_y}{2a} - \frac{\gamma-1}{2a}\frac{v}{a} & \frac{\hat{\xi}_z}{2a} - \frac{\gamma-1}{2a}\frac{w}{a} & \frac{\gamma-1}{2a^2} \end{bmatrix} \tag{2.3.64}$$

$$\mathbf{\Lambda}_\xi = \begin{bmatrix} \bar{u} - a & 0 & 0 & 0 & 0 \\ 0 & \bar{u} & 0 & 0 & 0 \\ 0 & 0 & \bar{u} & 0 & 0 \\ 0 & 0 & 0 & \bar{u} & 0 \\ 0 & 0 & 0 & 0 & \bar{u} + a \end{bmatrix} \tag{2.3.65}$$

where $\bar{u}$ is the dot product of the velocity vector with the unit vector normal to a cell face.

Many researchers currently use the flux-difference splitting technique described above. Further, Van Leer et al.[10] demonstrate that for viscous flows, this flux function is more accurate than central-difference formulations with scalar dissipation, as well as upwind formulations based on flux-vector splitting. The explanation lies in the consideration of Eq. (2.3.54) as a central-difference flux with an added dissipative term. By considering the influence of the individual waves, it is apparent that as an eigenvalue associated with the wave vanishes, the corresponding dissipation also vanishes. This mechanism is the means through which the exact solution to a single discontinuity is obtained, as shown in Eq. (2.3.55). For viscous flows, the boundary layer is considered to consist of a series of shear waves normal to the body. Because the velocity in this direction is small, the corresponding dissipation is also small; the result is that boundary layers are captured with very high accuracy.

Observe that vanishing eigenvalues (hence artificial viscosity for the wave) occur at shock waves, where $\tilde{\lambda}_k = \bar{u} - a$ passes through zero, as well as contact discontinuities, where $\tilde{\lambda}_k = \bar{u} = 0$. Unfortunately, vanishing eigenvalues also occur at sonic points at which the flow transitions from subsonic to supersonic flow and $\tilde{\lambda}_k = \bar{u} - a$ is again zero. This is a consequence of replacing the full nonlinear problem with a linearized version in Eq. (2.3.48) which considers an expansion to be described by a single wave instead of a series of waves. This can lead to expansion shocks in which the flow transitions from subsonic to supersonic in a discontinuous manner. To remedy the situation, the eigenvalue $\tilde{\lambda}_k = \bar{u} - a$ is often modified slightly so that it does not vanish at these points. As a result, a small amount of dissipation is added to the scheme so the wave is spread slightly over several mesh points. Although many forms of this modification exist, the most common implementation is to modify the eigenvalue according to[26]

$$|\tilde{\lambda}| \equiv \begin{cases} |\tilde{\lambda}|_{i+\frac{1}{2}} & \text{if } |\tilde{\lambda}|_{i+\frac{1}{2}} \geq \epsilon \\ \frac{1}{2}\left(\frac{\tilde{\lambda}^2_{i+\frac{1}{2}}}{\epsilon} + \epsilon\right) & \text{if } |\tilde{\lambda}|_{i+\frac{1}{2}} \leq \epsilon \end{cases} \tag{2.3.66}$$

An example of an expansion shock is shown in the lower portion of Fig. 2.3.7, which depicts the Mach number distribution through a quasi-one-dimensional converging/diverging nozel obtained with a first-order spatially accurate discretisation. As seen, without the smoothing of the eigenvalue, an expansion shock appears as the Mach number passes through unity. However, by restricting the value of $\tilde{\lambda}_k = \bar{u} - a$ to be nonzero, a smooth transition through the sonic point is obtained, and the expansion shock is eliminated. In practice, the addition of eigenvalue smoothing is often not required for higher order methods. However, several conditions, usually associated with very strong shocks, necessitate the inclusion of dissipation through the eigenvalue, similar to Eq. (2.3.66). Several researchers have independently observed these phenomena; an interesting summary of many of these conditions is given in Ref. 27.

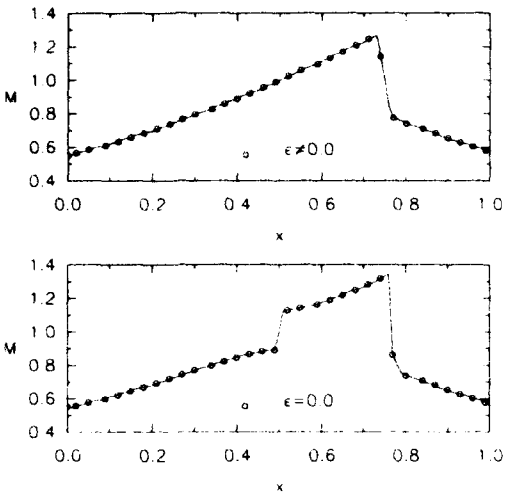


Figure 2.3.7 Example of expansion shock and the effectiveness of Eq. (2.3.66).

Several other flux functions have been developed in recent years that are not as widely utilized as those discussed above, but are nevertheless worthy of mention. The advection upstream splitting method (AUSM)[28] is a flux-vector splitting method that includes some of the favorable properties of flux-difference splitting while maintaining the computational efficiency of flux-vector splitting and is easier to linearize for use in implicit methods. It has been shown to capture contact discontinuities with accuracy similar to flux-difference splitting but does not suffer some of the deficiencies, such as expansion shocks. Two similar methodologies are the wave/particle split (WPS) method recently developed and reported in Ref. 29 and the convective upwind and split pressure (CUSP) scheme developed by Jameson.[6] Both of these schemes are similar to the AUSM scheme and exhibit similar accuracy. Several other variations to flux-vector splitting have also been derived and are discussed in Ref. 30. These include flux-splitting schemes based on higher order polynomials of the Mach number, similar to those of Liou and Steffen.[31] In addition, Coquel and Liou[32,33] have recently developed flux functions which combine the best features of both flux-vector splitting and flux-difference splitting by using flux-vector splitting for non-linear waves such as shocks, while using using a flux-difference splitting for the linear waves. This approach simultaneously preserves the robustness associated with flux-vector splitting as well as accuracy in capturing stationary contact discontinuities.

The characteristic flux extrapolation technique of Eberle has been used for computing flows about many practical geometries and is described in Refs. 34, 35, 36, 37, and 38. This method has been shown to be capable of capturing strong shocks and can also recover a constant total temperature. In Ref. 37, modifications to the flux-limiter of Van Albada[39] are also discussed, as well as the incorporation of equilibrium real gas.

Multidimensional Upwind Methods

The upwind techniques discussed above are applicable strictly to one-dimensional problems. To apply these techniques to two- and three-dimensional problems, the usual procedure is to assume that waves propagate normal to grid lines, which allow Riemann solvers to be applied in a one-dimensional manner separately in each coordinate direction. This approach leads to quite satisfactory solutions when features such as shock waves and shear waves are essentially aligned with the mesh. However, severe degradation of accuracy can occur when the features are oblique to the grid lines because they are interpreted by the Riemann solver to be composed of pairs of grid-aligned waves instead of a single wave. The result is that shocks and shear waves may be severely smeared. An illustration is given in Fig. 2.3.8 in which a single shear wave (dashed line) is misrepresented as a compression plus a shear wave because of the orientation of the grid (solid line). The consequence in this case is that an incorrect pressure difference across the wave is obtained, which may manifest itself in a computation as a pressure distortion.[40,41]

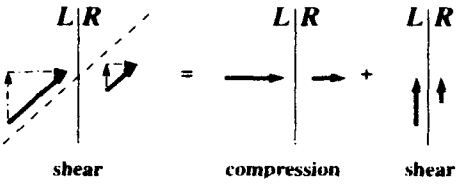


Figure 2.3.8 Representation of an oblique shear wave by two waves aligned with the grid.

In recent years, several research efforts have been aimed at overcoming the possible loss of accuracy attributable to the dimension-by-dimension approach to upwind differencing. A summary of some of the more promising techniques is given in Refs. 40 and 42.

In general, the approaches to multidimensional upwinding can be categorized into two groups: those based on computing a flux function and those that rely on fluctuation splitting. The first approach can further be divided into techniques that use a single dominant direction and those that use more than one direction. Flux-function methods that use a single dominant wave direction use a frame of reference aligned with the assumed wave direction to compute fluxes that are then rotated back into the grid-aligned frame. Examples of this type of approach include the work by Davis,[43] Dadone and Grossman,[44] Goorjian,[45] Hirsch et al.[46,47] and Levy, Powell, and Van Leer.[48] Generally, these methods show good improvement over grid-aligned methods for first-order accurate spatial differencing. For higher order methods, the improvements are less dramatic, and convergence to a steady state is sometimes difficult.[40]

Flux-function-based methods that utilize more than one wave direction include those of Rumsey et al.,[49,41] Parpia and Michalek,[50] and Powell, Barth, and Parpia.[51] In Refs. 41, 49, and 50, states are obtained on either side of a cell face in much the same manner as in the dimension-by-dimension approach. However the fluxes are computed by assuming that the states on either side of the face are connected by two acoustic waves and an entropy wave, all in the direction of the velocity difference, as well as a fourth (shear) wave that is normal to the first three. In Rumsey et al.,[49] the directions are usually frozen after a certain number of iterations to enhance

convergence. This requires that a fifth wave also be added because the four wave model can no longer correctly connect the states on either side of the face. The fifth wave is a weak shear wave that is assumed to be perpendicular to the velocity difference direction. In this way, the difference in states can be expressed as

$$\Delta \mathbf{w} = \sum_{k=1}^{5} \alpha_k \mathbf{R}_k \tag{2.3.67}$$

where α_k is the strength of the k^{th} wave and $\mathbf{R}_k$ is an eigenvector of the matrix $\mathbf{A}\cos\theta_k + \mathbf{B}\sin\theta_k$ ($\mathbf{A}$ and $\mathbf{B}$ are flux Jacobian matrices, and θ_k is the propagation angle of the k^{th} wave). The flux on the cell face is now computed by accounting for the influence of each wave as

$$\mathbf{F}_{i+\frac{1}{2}} = \frac{1}{2}(\mathbf{F}_L + \mathbf{F}_R) - \frac{1}{2}\sum_k (|\lambda_k|\Delta \mathbf{w})\mathbf{R}_k \tag{2.3.68}$$

In general, the success of schemes based on this methodology is similar to that described previously.

Multidimensional schemes based on fluctuation splitting are rapidly evolving. In general, these schemes are composed of three primary pieces:

1. A residual calculation method for the cell (typically a trapezoidal numerical integration of the fluxes)
2. A method for decomposing the cell residual into wave-like components (the "wave model")
3. A method for distributing changes caused by waves to the vertices of the cell, in such a way that the positivity of the resulting scheme is maintained.

Perhaps the most familiar fluctuation-splitting scheme is that of Ni[52] in which the Lax-Wendroff scheme is cast as a cell-vertex, or fluctuation-splitting method. More recent work in fluctuation splitting has improved this method in two important ways: the residual-distribution scheme (i.e., the method for "pushing" the changes to the vertices of the cell) has been improved with careful study of the scalar advection case[42,53,54,55,56] and various ways of breaking the residual into pieces due to planar waves (i.e., wave model) have been developed.[42,57] Progress on the residual-distribution schemes has been quite good, and positive, accurate methods for distributing the pieces of the residual have been developed and extensively tested. Many wave models have been developed to date, and the advantages and disadvantages of the various models have not been clearly established. Some of the differences among the wave models include the number of waves that are used (typically between four and six) and the directions in which the waves are assumed to propagate.

The fluctuation-splitting schemes have been shown to work quite well in supersonic flows, but issues of solution convergence and accuracy in subsonic flows have not been thoroughly addressed. Current work, such as that of Paillere et al.[58] is concentrated on these issues, and substantial progress is being made.

Higher Order Schemes

Up to this point in the discussion of upwind schemes, the determination of the left and right state variables on either side of a cell face has been, for the most part, left unspecified. Recall that for Godunov's method, if a piecewise constant approximation of the data in each cell is assumed, then the resulting scheme is first-order accurate in space. The accuracy of the approximation can be raised to higher order by replacing the piecewise constant approximation of the data with a piecewise polynomial approximation.[22,59] For instance, the state variables on the left and right side of the cell interface located at $i + 1/2$ in Fig. 2.3.9 can be determined as

$$\begin{aligned} \mathbf{w}_L &= \mathbf{w}_i + \frac{1}{4}[(1-\kappa)\Delta_- + (1+\kappa)\Delta_+]_i \\ \mathbf{w}_R &= \mathbf{w}_{i+1} - \frac{1}{4}[(1-\kappa)\Delta_+ + (1+\kappa)\Delta_-]_{i+1} \end{aligned} \tag{2.3.69}$$

where

$$\begin{aligned} (\Delta_+)_i &\equiv \mathbf{w}_{i+1} - \mathbf{w}_i \\ (\Delta_-)_i &\equiv \mathbf{w}_i - \mathbf{w}_{i-1} \end{aligned} \tag{2.3.70}$$

Equation (2.3.69) represents a one-parameter family of schemes. A fully one-sided approximation of the data is obtained by inserting $\kappa = -1$ while $\kappa = 1/3$ leads to a third-order upwind-biased approximation, and $\kappa = +1$ yields a second-order central-difference scheme. All the upwind-biased approximations use the same number of cells for the residual computation as the fully one-sided scheme and may be implemented with only a slight increase in computational effort. The third-order scheme is strictly third-order accurate only in one-dimensional calculations. To obtain a third-order scheme in two or three dimensions, computation of the flux across a cell face on the basis of an averaged state is not sufficient because the difference between that average flux and the flux computed from the averaged states is a term of second order and vanishes only for a linear system of conservation laws. Nevertheless, by switching from a fully upwind approximation ($\kappa = -1$) to the third-order ($\kappa = 1/3$) scheme, the accuracy of smooth solutions can be increased.[60]

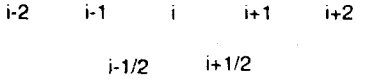


Figure 2.3.9 Higher order interpolation stencil.

A deficiency in using Eq. (2.3.69) for reconstructing the data at the cell faces is that new extrema can be introduced even when the original data is monotone. For example, in Fig. 2.3.10, a nonmonotone interpolation is obtained between cells $i-1$ and i. If this profile is convected and the cell averages

are then reconstructed, nonphysical oscillations can result in the solution.



Figure 2.3.10 Introduction of new extrema using upwind-biased interpolation.

For determining acceptable limits on the slopes, the data in each cell are first represented by a Taylor series expansion about the center of the cell. For example, the data on the boundaries of cell i in Fig. 2.3.10 can be determined as

$$\mathbf{w}_{i+\frac{1}{2}} = \mathbf{w}_i + \frac{1}{2}\left(\frac{\partial \mathbf{w}}{\partial \mathbf{x}}\right)\Delta \mathbf{x} = \mathbf{w}_i + \frac{1}{2}\delta \mathbf{w} \tag{2.3.71}$$

$$\mathbf{w}_{i-\frac{1}{2}} = \mathbf{w}_i - \frac{1}{2}\left(\frac{\partial \mathbf{w}}{\partial \mathbf{x}}\right)\Delta \mathbf{x} = \mathbf{w}_i - \frac{1}{2}\delta \mathbf{w} \tag{2.3.72}$$

where Δx is the width of the cell and

$$\delta \mathbf{w} = \frac{1}{2}[(1-\kappa)\Delta_- + (1+\kappa)\Delta_+], \tag{2.3.73}$$

For monotone increasing data, a sufficient bound on the size of Δ_+ and Δ_+ is obtained by requiring that the interpolated data on the cell faces does not exceed the values in the surrounding cells. This limit is achieved provided that

$$\begin{aligned} \mathbf{w}_i + \frac{1}{2}\delta \mathbf{w} \le \mathbf{w}_{i+1} &\Rightarrow \delta \mathbf{w} \le 2\Delta_+ \\ \mathbf{w}_i - \frac{1}{2}\delta \mathbf{w} \ge \mathbf{w}_{i-1} &\Rightarrow \delta \mathbf{w} \le 2\Delta_- \end{aligned} \tag{2.3.74}$$

In order to ensure a monotone interpolation, the magnitude of $\delta \mathbf{w}$ may have to be limited to be no larger than either $2\Delta_+$ or $2\Delta_-$:

$$(\delta \mathbf{w}) \le \min(2\Delta_+, 2\Delta_-) \tag{2.3.75}$$

Equation (2.3.75) provides a guideline for reducing the magnitude of any gradient that would result in a nonmonotone interpolation. Following Van Leer,[61] the value of $\delta \mathbf{w}$ that will maintain monotone interpolation will be referred to as $(\delta \mathbf{w})_{\text{limited}}$ and can be written as

$$(\delta \mathbf{w})_{\text{limited}} = R(o)\delta \mathbf{w} \tag{2.3.76}$$

where $o = \frac{\Delta_+}{\Delta_-}$ and $R(o)$ serves to limit the size of the original gradient. From Eqs. (2.3.73), (2.3.75), and (2.3.76), $R(o)$ is written as

$$R(o) = \min\left[\frac{4}{(1-\kappa)+(1+\kappa)o}, \frac{4o}{(1-\kappa)+(1+\kappa)o}\right] \tag{2.3.77}$$

In Fig. 2.3.11, a plot of $R(o)$ from Eq. (2.3.77) is shown, where κ has been assumed to be zero ($\kappa = 0$) in Eq. (2.3.73). In the figure, the area that lies inside the curve is the region

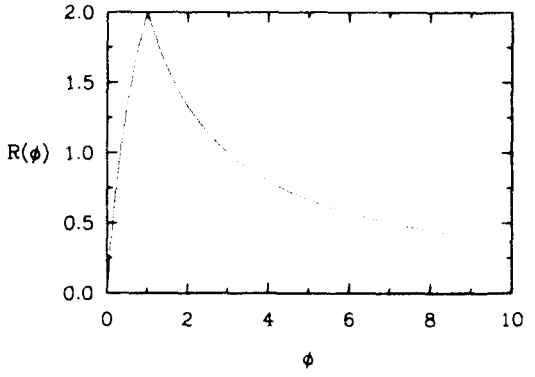


Figure 2.3.11 Boundary of $R(o)$ for monotone interpolation.

for which monotone interpolation is obtained.

For $o = 1$, Δ_+ is equal to Δ_-, and the data in the cell appears linear. Because a second-order scheme should reconstruct linear data exactly, a basic requirement on the limiter function is that it pass through unity for $o = 1$, which can easily be achieved by modifying Eq. (2.3.77) to read

$$R(o) = \min\left[\frac{4}{(1-\kappa)+(1+\kappa)o}, 1, \frac{4o}{(1-\kappa)+(1+\kappa)o}\right] \tag{2.3.78}$$

In this manner, the slopes as calculated from Eq. (2.3.73) are left unchanged, provided the interpolations remain monotone.

Many variations for $R(o)$ have appeared in the literature that preserve monotonicity and are second-order accurate. One of the most commonly used limiters is developed by Van Albada[39] and is given by

$$R(o) = \frac{2o}{(o^2+1)} \tag{2.3.79}$$

while a limiter function of Van Leer is given by[62]

$$R(o) = \frac{4o}{(o+1)^2} \tag{2.3.80}$$

Note that as long as $R(o)$ maintains second-order accuracy and remains within the monotone region, $R(o)$ does not need to remain below 1 in magnitude. Therefore, the limiter function can actually serve to increase the slope calculated from Eq. (2.3.73). An example of such a limiter is Roe's "superbee" limiter[63]

$$R(o) = \begin{cases} \frac{1o}{(1+o)} & 0 \le o < \frac{1}{3} \\ \frac{2}{(1+o)} & \frac{1}{2} \le o < 1 \\ \frac{2o}{(1+o)} & 1 \le o \le 2 \\ \frac{1}{(1+o)} & o > 2 \end{cases} \tag{2.3.81}$$

A plot of the Van Albada, Van Leer, and Roe's superbee limiters is shown in Fig. 2.3.12. Note that all the limiters

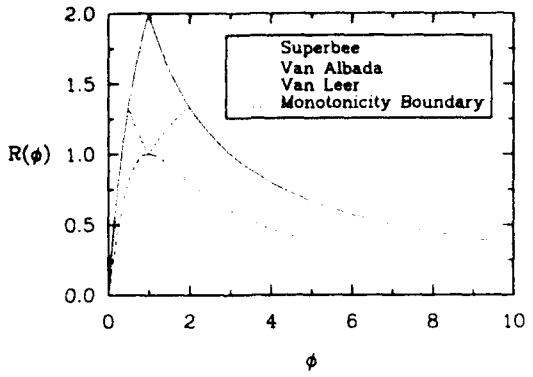


Figure 2.3.12 Superbee, Van Albada, and Van Leer limiters.

shown pass through unity when $\phi = 1$ which maintains second-order accuracy.

In the previous discussion, it has been assumed that the data are monotonically increasing: similar arguments hold for monotonically decreasing data. For nonmonotone data in which the sign of Δ_+ and Δ_- differ, $(\delta \mathbf{w})_{\text{limited}}$ can be simply set to zero to make sure that any extrema are not magnified.[61] In the work of Spekreijse,[64,65] general conditions are derived for the limiter function that will maintain second-order accuracy as well as monotone steady-state solutions that do not set the gradient to zero at the extrema.

Another method that is useful to the design of nonoscillatory schemes is based on the definition of total variation. For a discrete one-dimensional scalar solution on an infinite domain, the total variation is defined at time level n as

$$TV(W^n) = \sum_{-\infty}^{\infty} |W_{i+1} - W_i| \tag{2.3.82}$$

In the top part of Fig. 2.3.13, for a monotone grid function, the total variation is determined strictly by the endpoints (i.e. $|W^n_\infty - W^n_{-\infty}|$). However, if a new extrema is introduced as in the lower half of the figure, the total variation will increase. Hence, for a scheme to remain nonoscillatory, the total variation should remain the same or decrease as the solution is updated:

$$TV(W^{n+1}) \le TV(W^n) \tag{2.3.83}$$

Schemes derived from these guidelines are appropriately called total variation diminishing (TVD).

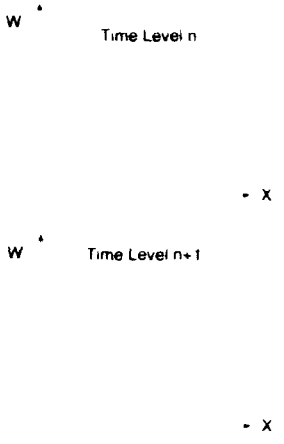


Figure 2.3.13 Example of increasing total variation.

Sufficient conditions for constructing TVD schemes were first developed by Harten.[66] While many investigators have examined criteria for constructing TVD schemes, (e.g., 67 and 68), the method of analysis given by Barth in reference 69 relies upon a matrix interpretation and is described below.

For obtaining bounds on the variation, a general form for a conservative difference scheme is first written as

$$W_i^{n+1} + \sum_{j=-p}^{q} D(j)_{i+\frac{1}{2}} \Delta_{i+\frac{1}{2}+j} W^{n+1} = W_i^n + \sum_{j=-p}^{q} C(j)_{i+\frac{1}{2}} \Delta_{i+\frac{1}{2}+j} W^n \tag{2.3.84}$$

where $D(j)_{i+\frac{1}{2}}$ and $C(j)_{i+\frac{1}{2}}$ are nonlinear functions of W at the grid points and

$$\Delta_{i+\frac{1}{2}} W = W_{i+1} - W_i \tag{2.3.85}$$

By assuming a periodic domain, the definition of the discrete total variation can be expressed in terms of a difference matrix $\mathbf{H}$ as

$$\mathbf{H} = \begin{bmatrix} -1 & 1 & 0 & 0 & \cdots & 0 \\ 0 & -1 & 1 & 0 & \cdots & 0 \\ 0 & 0 & -1 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & -1 & 1 \\ 1 & 0 & 0 & 0 & 0 & -1 \end{bmatrix} \tag{2.3.86}$$

With Eq. (2.3.86), the total variation can be written as

$$TV(W) = \|\mathbf{H}W\|_1 \tag{2.3.87}$$

where $\|\mathbf{v}\|_1$ is the L_1 vector norm given as

$$\|\mathbf{v}\|_1 = \sum_j |v_j| \tag{2.3.88}$$

The TVD condition can then be expressed as

$$\|\mathbf{H}W^{n+1}\|_1 \le \|\mathbf{H}W^n\|_1 \tag{2.3.89}$$

The general difference scheme, Eq. (2.3.84), can be rewritten with the forward-difference matrix as

$$\left[\mathbf{I}+\theta\widetilde{\mathbf{M}}\mathbf{H}\right]\mathbf{W}^{n+1}=[\mathbf{I}-(1-\theta)\mathbf{M}\mathbf{H}]\mathbf{W}^{n} \tag{2.3.90}$$

where $\widetilde{\mathbf{M}}$ and $\mathbf{M}$ are matrix operators that may be nonlinear functions of $\mathbf{W}$ because the C's and D's in Eq. (2.3.84) are themselves functions of $\mathbf{W}$. For $\theta = 0$, a fully explicit method is obtained and implicit formulations are otherwise obtained.

Multiplying Eq. (2.3.90) from the left by the forward-difference matrix $\mathbf{H}$ and regrouping terms yields an equation for the evolution of the variation

$$\begin{aligned}\mathcal{L}(\mathbf{H}\mathbf{W}^{n+1})&=\mathcal{R}(\mathbf{H}\mathbf{W}^{n})\Rightarrow\\ \mathbf{H}\mathbf{W}^{n+1}&=\mathcal{L}^{-1}\mathcal{R}(\mathbf{H}\mathbf{W}^{n})\end{aligned} \tag{2.3.91}$$

where

$$\mathcal{L}=\left[\mathbf{I}+\theta\mathbf{H}\widetilde{\mathbf{M}}\right] \tag{2.3.92}$$

$$\mathcal{R}=[\mathbf{I}-(1-\theta)\mathbf{H}\mathbf{M}] \tag{2.3.93}$$

By taking the L_1 matrix norm of Eq. (2.3.91)

$$\begin{aligned}\|\mathbf{H}\mathbf{W}^{n+1}\|_1&=\|\mathcal{L}^{-1}\mathcal{R}(\mathbf{H}\mathbf{W}^{n})\|_1\\ &\le\|\mathcal{L}^{-1}\mathcal{R}\|_1\|\mathbf{H}\mathbf{W}^{n}\|_1\\ &\le\|\mathcal{L}^{-1}\|_1\|\mathcal{R}\|_1\|\mathbf{H}\mathbf{W}^{n}\|_1\end{aligned} \tag{2.3.94}$$

a sufficient condition for the scheme to remain TVD is $\|\mathcal{L}^{-1}\mathcal{R}\|_1 \le 1$. In addition, because $\|\mathcal{L}^{-1}\mathcal{R}\|_1 \le \|\mathcal{L}^{-1}\|_1\|\mathcal{R}\|_1$, if both $\|\mathcal{L}^{-1}\|_1 \le 1$ and $\|\mathcal{R}\|_1 \le 1$, the resulting scheme will be TVD as well.

By first considering the explicit operator $\mathcal{R}$, one can sum the elements in the columns by multiplying on the left by a row matrix $\mathbf{s} = [1, 1, \ldots, 1]$. Now, because $\mathbf{s}\mathbf{H} = [0, 0, \ldots, 0]$, Eq. (2.3.93) shows that the columns of $\mathcal{R}$ will sum to unity regardless of the choice of $\mathbf{M}$. The L_1 matrix norm corresponds to summing the absolute value of each of the columns and taking the largest of these values. Because each column sums to unity, a necessary and sufficient condition for $\|\mathcal{R}\|_1 \le 1$ is that each element of $\mathcal{R}$ be nonnegative. Otherwise, when the absolute value of the individual elements is taken to compute the norm, the sum would be greater than unity (i.e. $\|\mathcal{R}\|_1 \ge 1$).

Application of this criteria to explicit schemes results in identical requirements given originally by Harten,[66] as can be seen by considering a general explicit scheme written at a node point j in the form

$$\begin{aligned}W_j^{n+1}=W_j^{n}&+C^{+}_{j-\frac{1}{2}}(W_{j+1}^{n}-W_j^{n})\\ &-C^{-}_{j-\frac{1}{2}}(W_j^{n}-W_{j-1}^{n})\end{aligned} \tag{2.3.95}$$

By subtracting the expression at node j from that at j+1, an expression for the evolution of $\mathbf{H}W^{n}$ at node point j can be obtained:

$$\begin{aligned}\mathbf{H}W_j^{n+1}=&\left[\mathbf{I}-C^{+}_{j+\frac{1}{2}}-C^{-}_{j-\frac{1}{2}}\right]\mathbf{H}W_j^{n}\\ &+C^{+}_{j+\frac{3}{2}}\mathbf{H}W_{j+1}^{n}+C^{+}_{j-\frac{1}{2}}\mathbf{H}W_{j-1}^{n}\end{aligned} \tag{2.3.96}$$

If this expression is written at each mesh point and assembled into a matrix, then

$$\mathcal{R}=\begin{bmatrix}\ddots & & 0 & 0 & \\ & \ddots & C^{+}_{\cdots} & 0 & 0\\ 0 & C^{-}_{\cdots} & 1-C^{+}_{\cdots}-C^{-}_{\cdots} & C^{+}_{\cdots} & 0\\ 0 & 0 & C^{-}_{\cdots} & \ddots & \\ & 0 & 0 & & \ddots\end{bmatrix} \tag{2.3.97}$$

Now, if each element of this matrix is required to be positive, then the TVD criteria given by Harten in Ref. 66 results:

$$\begin{aligned}&1-C^{+}_{\cdots}-C^{-}_{\cdots}\ge 0\\ &C^{+}_{j+\frac{1}{2}}\ge 0\\ &C^{-}_{j+\frac{1}{2}}\ge 0\end{aligned} \tag{2.3.98}$$

A similar procedure can be used for implicit schemes and is not repeated here.

Many schemes have been developed and applied to difficult aerodynamic problems that rely heavily on the TVD concepts discussed above. While by no means complete, a few examples can be found in Refs. 24, 26, 70, 71, and 72, as well as in the references contained therein.

The methodologies described above only achieves higher order accuracy for smoothly varying meshes. To obtain higher order accuracy on general meshes, as well as to extend the accuracy beyond second order, an extensive amount of research has been conducted for both structured and unstructured grids. An important class of such schemes is the essentially nonoscillatory (ENO) schemes. Although no details are given here, the essential ingredients include a polynomial reconstruction of the data from cell-averaged data that is exact up to a specified order of accuracy:

$$R(x;\overline{\mathbf{w}})=\mathbf{w}(x)+O(h^{r}) \tag{2.3.99}$$

where $R(x;\overline{\mathbf{w}})$ represents the data reconstructed from the cell averages and $\mathbf{w}(x)$ is the exact value. The scheme also must be conservative in the sense that if one integrates the reconstructed data over the cell, then the cell average is recovered. The last criteria is that the reconstructed data be "essentially nonoscillatory." That is,

$$TV(R(x;\overline{\mathbf{w}}))\le TV(\mathbf{w}(x))+O(h^{r}) \tag{2.3.100}$$

In order to satisfy Eq. (2.3.100) in a smooth region of a discontinuous solution, the stencil used in the reconstruction process is varied as the calculation proceeds. Further details and descriptions of the theory, as well as applications of this type of technique, can be found in many references.[73,74,75,76,77]

Another technique to obtain higher order accuracy that has been particularly useful for unstructured grids is the so-called "k-exact" method. Here, the conservation of the mean is enforced, and the reconstruction is such that a polynomial of degree k or less is reconstructed exactly.[78] In the implementations of Refs. 77, 79, and 80, the stencil generally remains

fixed which results in a somewhat lower computational expense. However, in order to avoid oscillations, a limiting procedure must be applied where steep gradients are present in order to avoid nonphysical oscillations. This tends to reduce the order of accuracy in these regions. More recently, algorithms have been proposed that incorporate stencil-varying techniques into high-order ENO methods for unstructured grids.[81,82]

High-order ENO schemes can also be implemented in a finite-difference manner in which the reconstruction operator acts upon pointwise fluxes.[83]

2.3.3 Extension to Real Gases

Many of the flux functions discussed above have been extended for use with real gases, including both equilibrium and finite-rate chemistry. In Ref. 84, for example, the flux-splitting schemes of Steger-Warming[12] and Van Leer[17] are extended for real-gas considerations as well as for the flux-difference splitting of Roe.[25] Alternate derivations that do not rely on some of the assumptions made in the previous reference are given for an equilibrium real gas in Ref. 85 for flux-vector splitting and in Ref. 86 for Roe's flux-difference splitting scheme. Additional real-gas extensions are given in Refs. 87 and 88. For the characteristic flux extrapolation technique of Eberle,[47] real-gas extensions have also been made and are reported in Refs. 34, 37, and 89.

2.3.4 References

1 RADESPIEL, R., ROSSOW, C., and SWANSON, R. C., "An Efficient Cell-Vertex Multigrid Scheme for the Three-Dimensional Navier-Stokes Equations," AIAA 89–1953, June 1989.

2 VATSA, V. N., and WEDAN, B. W., "Development of an Efficient Multigrid Code for 3–D Navier-Stokes Equations," AIAA 89–1791, June 1990.

3 MARTINELLI, L., *Calculation of Viscous Flows with Multigrid Methods*. PhD thesis, Princeton University, 1987.

4 PULLIAM, T. J., "Artificial Dissipation Models for the Euler Equations," AIAA 85–0438, 1985.

5 JAMESON, A., SCHMIDT, W., and TURKEL, E., "Numerical Solutions of the Euler Equations by Finite Volume Methods Using Runge-Kutta Time-Stepping Schemes," AIAA 81–1259, 1981.

6 JAMESON, A., "Computational Algorithms for Aerodynamic Analysis and Design," MAE Report 1966 Princeton University, 1992.

7 SWANSON, R. C., and TURKEL, E., "Artificial Dissipation and Central Difference Schemes for the Euler and Navier-Stokes Equations," AIAA 87–1107, June 1987.

8 TURKEL, E., "Improving the Accuracy of Central Difference Schemes," in *11th International Conference on Numerical Methods in Fluid Dynamics, Springer-Verlag Lecture Notes in Physics*, vol. 323, pp. 586–591, 1988.

9 SWANSON, R. C., and TURKEL, E., "On Central-Difference and Upwind Schemes," J. Comp. Phys., vol. 101, pp. 292–306, 1992. Also ICASE Report 90–44 1990.

10 VAN LEER, B., THOMAS, J., ROE, P., and NEWSOME, R., "A Comparison of Numerical Flux Formulas for the Euler and Navier-Stokes Equations," AIAA 87–1104, 1987.

11 VATSA, V. N., THOMAS, J. L., and WEDAN, B. W., "Navier-Stokes Computations of Prolate Spheroids at Angle of Attack," AIAA 87–2627–CP, Aug. 1987.

12 STEGER, J. L., and WARMING, R. F., "Flux Vector Splitting of the Inviscid Gasdynamic Equations with Application to Finite-Difference Methods," J. Comp Phys., vol. 40, pp. 263–293, Apr. 1981.

13 JANUS, J. M., "The Development of a Three-Dimensional Split Flux Vector Euler Solver With Dynamic Grid Applications," Master's thesis, Mississippi State University, Aug. 1984.

14 BUNING, P. G., and STEGER, J. L., "Solution of the Two-Dimensional Euler Equations with Generalized Coordinate Transformation Using Flux Vector Splitting," AIAA 82–0971, June 1982.

15 VON LAVANTE, E., CLAES, D., and ANDERSON, W. K., "The Effects of Various Implicit Operators on a Flux Vector Splitting Method," AIAA 86–0273, Jan. 1986.

16 JESPERSEN, D. C., and PULLIAM, T. H., "Flux Vector Splitting and Approximate Newton Methods," AIAA 83–1899, July 1983.

17 B. VAN LEER, "Flux Vector Splitting for the Euler Equations," Lecture Notes in Physics, vol. 170, pp. 501–512, 1982.

18 ANDERSON, W. K., THOMAS, J. L., and VAN LEER, B., "A Comparison of Finite Volume Flux Vector Splittings for the Euler Equations," AIAA Journal, vol. 24, pp. 1453–1460, Sept. 1986.

19 HaNEL, D., SCHWANE, R., and SEIDER, G., "On the Accuracy of Upwind Schemes for the Solution of the Navier-Stokes Equations," AIAA 87–1105, 1987.

20 HIRSCH, C., *Numerical Computation of Internal and External Flows*. Wiley, 1990.

21 VAN LEER, B., "Towards the Ultimate Conservative Difference Scheme. III: Upstream-Centered Finite Difference Schemes for Ideal Compressible Flow," J. Comp. Phys., vol. 23, pp. 263–275, 1977.

22 VAN LEER, B., "Towards the Ultimate Conservative Difference Scheme. V: A Second Order Sequel to Godunov's Method," J. Comp. Phys., vol. 32, pp. 101–136, 1979.

23 OSHER, S., and SOLOMON, F., "Upwind Difference Schemes for Hyperbolic Systems of Conservation Laws," Mathematics of Computations, vol. 38, no. 158, pp. 339–374, 1982.

24 CHAKRAVARTHY, S. R., "Development of Upwind Schemes for the Euler Equations," NASA Contractor Report 4043, 1987.

25 ROE, P., "Approximate Riemann Solvers, Parameter Vectors, and Difference Schemes," J. of Comp. Phys., vol. 43, pp. 357–372, 1981.

26 HARTEN, A., "On a Class of High Resolution Total-Variation-Stable Finite-Difference Schemes," SIAM J. Numer. Anal., vol. 21, Feb. 1984.

27 QUIRK, J. J., "A Contribution to the Great Riemann Solver Debate," ICASE Report 92–64, 1992.

28 LIOU, M., and STEFFEN, C. J., "A New Flux Splitting Scheme," NASA Technical Memorandum 104452, 1991.

29 HALT, D. W., and AGARWAL, R. K., "A Novel Algorithm for the Solution of Compressible Euler Equations in Wave/Particle (WPS) Form," AIAA 93–3392 CP, 1993.

30 COIRIER, W. J., and VAN LEER, B., "Numerical Flux Formulas for the Euler and Navier-Stokes Equations II. Progress in Flux-Vector Splitting," AIAA 91–1566–CP, 1991.

31 LIOU, M. S., and STEFFEN, C., "High-Order Polynomial Expansions for Flux-Vector Splitting," Paper presented at the International Conference on Computational Engineering Science, 1991.

32 COQUEL, F., and LIOU, M., "Field by Field Hybrid Upwind Splitting Methods," AIAA 93–3302–CP, 1993.

33 COQUEL, F., and LIOU, M., "Stable and Low Diffusive Hybrid Upwind Splitting Methods," in *First European CFD Conference* (Hirsh, C., Periaux, J., and Onate, E., eds.), vol. 1, pp. 9–16, 1992.

34 EBERLE, A., "Characteristic Flux Averaging Approach to the Solution of Euler's Equations," VKI Lecture Series 1987–04, 1987.

35 EBERLE, A., "3D Euler Calculations Using Characteristic Flux Extrapolation," AIAA 85–0119, 1985.

36 EBERLE, A., and HEISS, S., "Enhanced Numerical Inviscid and Viscous Fluxes for Cell Centered Finite Volume Schemes," Computers and Fluids, vol. 22, no. 2/3, pp. 295–309, 1993.

37 EBERLE, A., SCHMATZ, M., and BISSINGER, N., "Generalized Flux Vectors for Hypersonic Shock-Capturing," AIAA 90–0390, 1990.

38 HEISS, S., EBERLE, A., FORNASIER, L., and PAUL, W., "Application of the Euler Method EUFLEX to a Fighter-Type Airplane Configuration at Transonic Speed," AIAA 92–2620, 1992.

39 VAN ALBADA, G. D., VAN LEER, B., and ROBERTS, W. W., "High Resolution Schemes Usng Flux Limiters For Hyperbolic Conservation Laws," J. Astron. Astroph., vol. 108, 1982.

40 VAN LEER, B., "Progress in Multi-Dimensional Upwind Differencing," ICASE Report 92–43, 1992.

41 RUMSEY, C. L., *Development of a Grid-Independent Approximate Riemann Solver*, PhD thesis, University of Michigan, 1991.

42 STRUIJS, R., DECONINCK, H., DEPALMA, P., ROE, P., and POWELL, K., "Progress on Multidimensional Upwind Euler Solvers for Unstructured Grids," AIAA 91–1550–CP, 1991.

43 DAVIS, S. F., "A Rotationally-Biased Upwind Difference Scheme for the Euler Equations," J. Comp. Phys., vol. 56, 1984.

44 DADONE, A., and GROSSMAN, B., "A Rotated Upwind Scheme for the Euler Equations," AIAA paper 91–0635, 1991.

45 GOORJIAN, P. M., "A New Algorithm for the Navier-Stokes Equations Applied to Transonic Flows over Wings," AIAA 8th Computational Fluid Dynamics Conference, 1987.

46 HIRSCH, C., LACOR, C., and DECONINCK, H., "Convection Algorithms Based on a Diagonalization Procedure for the Multidimensional Euler Equations," AIAA 87–1163, 1987.

47 HIRSCH, C., and LACOR, C., "Upwind Algorithms Based on a Diagonalization of the Multidimensional Euler Equations," AIAA 89–1958, 1989.

48 LEVY, D., POWELL, K. G., and VAN LEER, B., "Implementation of a Grid-Independent Upwind Scheme for the Euler Equations," AIAA 89–1931CP, 1989.

49 RUMSEY, C. L., VAN LEER, B., and ROE, P. L., "A Multidimensional Flux Function with Applications to the Euler and Navier-Stokes Equations," J. Comp. Phys., vol. 105, no. 2, 1993.

50 PARPIA, I., and MICHALEK, D., "A Shock Capturing Method for Multidimensional Flow," AIAA 90–3016, 1990.

51 POWELL, K. G., BARTH, T. J., and PARPIA, I. F., "A Solution Scheme for the Euler Equations Based on a Multidimensional Wave Model," AIAA 93–0065, 1993.

52 NI, R.-H., "A Multiple-Grid Scheme for Solving the Euler Equations," AIAA Journal, vol. 20, Nov. 1982.

53 DECONINCK, H., STRUIJS, R., BOURGOIS, G., PAILLERE, H., and ROE, P. L., "Multidimensional Upwind Methods for Unstructured Grids," in AGARD report 787 *Unstructured Grid Methods for Advection Dominated Flows*, 1992.

54 ROSSOW, C. C., "Accurate Solution of the 2-D Euler Equations with an Efficient Cell-Vertex Upwind Scheme," AIAA 93–0071, 1993.

55 SIDILKOVER, D., *Numerical Solution to Steady-State Problems with Discontinuities*, PhD thesis, Weizmann Institute of Science, 1989.

56 ROE, P. L., and SIDILKOVER, D., "Optimum Positive Linear Schemes for Advection in Two and Three Dimensions," Submitted to J. of Comp. Phys., 1991.

57 ROE, P. L., and BEARD, L., "An Improved Wave Model for Multidimensional Upwinding of the Euler Equations," Proceedings of the 13th International Conference on Numerical Fluid Dynamics, 1992.

58 PAILLERE, H., DECONINCK, H., STRUIJS, R., ROE, P., and MüLLER, J.-D., "Computation of Inviscid Compressible Flows Using Fluctuation Splitting on Triangular Meshes," AIAA 93–3301, 1993.

59 COLELLA, P., and WOODWARD, P. R., "The Piecewise Parabolic Method (PPM) for Gas-Dynamical Simulations," J. Comp. Phys., vol. 54, pp. 174–201, 1984.

60 THOMAS, J. L., and WALTERS, R. W., "Upwind Relaxation Algorithms for the Navier Stokes Equations," AIAA 85–1501, July 1985.

61 VAN LEER, B., "Numerical Fluid Dynamics II," ICASE Internal Report, no. 36, 1987.

62 VAN LEER, B., "Towards the Ultimate Conservative Difference Scheme. II: Monotonicity and Conservation Combined in a Second Order Scheme," J. Comp. Phys., vol. 14, pp. 361–370, 1974.

63 SWEBY, P. K., "High Resolution Schemes Using Flux Limiters For Hyperbolic Conservation Laws," SIAM J. Numer. Anal., vol. 21, pp. 995–1011, Aug. 1984.

64 SPEKREIJSE, S., "Multigrid Solution of the Steady Euler Equations," CWI-tract 46 (Center for Mathematics and Computer Science, Amsterdam), 1988.

65 SPEKREIJSE, S. P., "Multigrid Solution of Monotone Second-Order Discretizations of Hyperbolic Conservation Laws," Math. Comp., vol. 49, pp. 135–155, 1987.

66 HARTEN, A., "High Resolution Schemes for Hyperbolic Conservation Laws," J. of Comp. Phys., vol. 49, pp. 357–393, 1983.

67 OSHER, S., and CHAKRAVARTHY, S., "Very High Order Accurate TVD Schemes," Tech. Rep. 84–44, ICASE, Sept. 1984.

68 JAMESON, A., and LAX, P., "Conditions for the Construction of Multi-Point Total Variation Diminishing Difference Schemes," Tech. Rep. 178076, ICASE, Sept. 1986.

69 BARTH, T. J., "Numerical Aspects of Computing Viscous High Reynolds Number Flows on Unstructured Meshes," AIAA 91-0721, 1991.

70 CHAKRAVARTHY, S. R., and OSHER, S., "A New Class of High Accuracy TVD Schemes for Hyperbolic Conservation Laws," AIAA 85-0363, 1985.

71 YEE, H. C., "Construction of Explicit and Implicit Symmetric TVD Schemes and Their Applications," J. Comp. Phys., vol. 68, pp. 151–179, 1987.

72 JAMESON, A., "A Non-Oscillatory Shock Capturing Scheme Using Flux Limited Dissipation," in *Lectures in Applied Mathematics* (Engquist, B. E., Osher, S., and Sommerville, R. C. J., eds.), 1985. Also MAE Report 1653 Princeton University.

73 HARTEN, A., ENGQUIST, B., OSHER, S., and CHAKRAVARTHY, S. R., "Uniformly High Order Accurate Essentially Non-Oscillatory Schemes III," ICASE Report 86–22, 1986.

74 CHAKRAVARTHY, S. R., HARTEN, A., and OSHER, S., "Essentially Non-Oscillatory Shock-Capturing Schemes of Arbitrarily High Accuracy," AIAA 86-0339, 1986.

75 CHAKRAVARTHY, S. R., HARTEN, A., and OSHER, S., "A New Class of High Accuracy TVD Schemes for Hyperbolic Conservation Laws," AIAA 86-0363, 1986.

76 CASPER, J., and ATKINS, H., "A Finite-Volume High-Order ENO Scheme for Two-Dimensional Hyperbolic Systems," J. Comp. Phys., vol. 106, pp. 62–76, May 1993.

77 GODFREY, A. G., *Topics on Spatially High-Order Accurate Methods and Preconditioning for the Navier-Stokes Equations with Finite-Rate Chemistry*, PhD thesis, Virginia Polytechnic Institute and State University, 1992.

78 BARTH, T. J., "Aspects of Unstructured Grids and Finite-Volume Solvers for the Euler and Navier-Stokes Equations," in AGARD Report 787 Unstructured Grid Methods for Advection Dominated Flows, 1992.

79 BARTH, T., and FREDERICKSON, P., "Higher Order Solution of the Euler Equations on Unstructured Grids Using Quadratic Reconstruction," AIAA 90-0013, Jan. 1990.

80 MITCHELL, C. R., and WALTERS, R. W., "K-Exact Reconstruction for the Navier Stokes Equations on Arbitrary Grids," AIAA 93-0536, 1993.

81 HARTEN, A., and CHAKRAVARTHY, S., "Multidimensional ENO Schemes for General Geometries," NASA Contractor Report 187637, also ICASE Report 91-76, 1991.

82 ABGRALL, R., "Design of an Essentially Non-Oscillatory Reconstruction Procedure on Finite-Element Type Meshes," ICASE Report 91-84, Dec. 1991.

83 SHU, C., and OSHER, S., "Efficient Implementation of Essentially Non-Oscillatory Shock-Capturing Schemes," J. Comp. Phys., vol. 77, no. 2, pp. 439–471, 1988.

84 GROSSMAN, B., and WALTERS, R. W., "An Analysis of Flux-Split Algorithms for Eulers Equations with Real Gases," AIAA paper 87-1117, 1987.

85 VINOKUR, M., "Generalized Flux-Vector Splitting for an Equilibrium Real Gas," NASA Contractor Report 177513, 1988.

86 VINOKUR, M., "Flux Jacobian Matrices and Generalized Roe average for an Equilibrium Real Gas," NASA Contractor Report 177512, 1988.

87 GLAISTER, P., "An Approximate Linearised Riemann Solver for the Euler Equations for Real gases," J. Comp. Phys., vol. 74, pp. 382–408, 1988.

88 LIOU, M., VAN LEER, B., and SHUEN, J. S., "Splitting of Inviscid Fluxes for Real Gases," NASA TM 100856, Apr. 1988.

89 SCHMATZ, M. A., "Hypersonic 3D Navier-Stokes Calculations for Equilibrium Gas," AIAA 89-2183, 1989.

2.4 TIME DISCRETIZATION

A number of different explicit and implicit schemes which have been used to compute three-dimensional solutions to the Euler equations are reviewed below. The general process of discretizing the governing flow equations leads to a system of equations of the form

$$\mathbf{H}(\mathbf{w}) = 0 \tag{2.4.1}$$

where $\mathbf{w}$ is the vector of unknowns at all grid points for a node-based scheme and at elements for a cell-based formulation. The solution of these equations can be obtained by using an explicit or implicit formulation for $\mathbf{w}$. Algorithms of an explicit formulation can be written as

$$\mathbf{w}^{n+1} = \hat{\mathbf{H}}(\mathbf{w}^n) \tag{2.4.2}$$

while an implicit scheme can be expressed as

$$\hat{\mathbf{H}}(\mathbf{w}^{n+1}, \mathbf{w}^n) = 0 \tag{2.4.3}$$

Here, $\mathbf{w}^n$ can be interpreted as the value of $\mathbf{w}$ at either time level or iteration level n; the particular form of $\hat{\mathbf{H}}$ arises from the type of time discretization used. Most of the applications to date have been for steady flows, in which the role of the time discretization is to facilitate or accelerate convergence to steady state. A requirement also exists for time-accurate simulations of the unsteady flow fields associated with dynamic vehicle motions or time-dependent fluid motions arising from separated and/or vortical flow fields.

The time discretization methods used to solve the Euler equations can be typed into two classes: coupled space-time methods and semidiscrete algorithms. In the latter approach, the spatial discretization is decoupled from the temporal discretization by first differencing the spatial derivative terms; the partial differential equations are thus transformed to a system of first-order ordinary differential equations in time. For steady flows, the time-rate-of-change of the spatial residual equation can be driven to zero, and the resulting physical solution is independent of the particular path taken to convergence or the time step used to advance the equations. As a consequence, the solution is only dependent on the spatial differencing approximation.

Explicit schemes (predictor-corrector or Runge-Kutta, for example) are simpler techniques than implicit schemes and lend themselves to extremely efficient implementations on either vector or parallel processing computers. The explicit schemes have a time-step limitation, corresponding to a Courant-Friedrichs-Levy (CFL) number on the order of unity, in order to maintain numerical stability of the solution. The time step for a CFL number of unity scales on the distance between the grid points; this time step limitation generally leads to inefficient simulations of unsteady flows corresponding to low reduced frequencies. In these cases, the time step required for accurate resolution of the time-dependent phenomena (generally associated with a fixed number of time steps per cycle) can be much larger than the allowable time step based on stability.

One of the motivations for developing an implicit Euler solver is to serve as a vehicle for solving the viscous equations. For a diffusion-dominated flow, the allowable explicit time step scales on the square of the distance between the grid points; for the highly clustered grids required for the resolution of viscous flows at high Reynolds numbers, the maximum time step of purely explicit schemes is prohibitively small. Implicit schemes have a less restrictive time step limitation and are generally more versatile and efficient, especially for time-dependent computations. However, the implicit schemes entail more arithmetic operations, since the solution of a coupled system of equations is generally required.

The implicit methods discussed below are restricted to the class of semidiscrete algorithms. Direct methods, which have been used to solve the accompanying large banded system of linear equations in a fixed number of operations, are contrasted with approximate factorization methods, which have been used to approximate the linear system as a product of simpler and more easily invertible matrices. The factorizations discussed, which rely on a regular ordering of the grid, have been widely used in structured-grid applications. Iterative techniques are then discussed, including relaxation, hybrid relaxation-factorization approaches, and minimum-residual methods, such as conjugate-gradient or Krylov-subspace methods. The direct methods and the iterative techniques can be generalized for use as implicit solvers for unstructured-grid schemes. Convergence acceleration techniques, such as multigrid, can be applied to either the linear or nonlinear implicit schemes above, as well as to explicit schemes, and are discussed in a subsequent section.

2.4.1 Explicit Schemes

To illustrate the explicit procedures, consider the model problem

$$\frac{\partial u}{\partial t} + f = 0 \tag{2.4.4}$$

where $u=u(x,t)$ and $f = f\left(\frac{\partial u}{\partial x}, u, x, t\right)$. Equations of this type arise in many branches of continuum mechanics, for example, one-dimensional unsteady fluid flow, where the equation

$$\frac{\partial u}{\partial t} + \frac{\partial}{\partial x}(\rho u) = 0 \tag{2.4.5}$$

represents conservation of mass.

In the application of the finite difference method, differences must be taken with respect to time t and distance x. A grid is constructed over the (x,t) plane and the equation is solved at each mesh point or cell. A typical mesh point can be represented as $(x,t^n) = (j\Delta x, n\Delta t)$. The value of u at this mesh point is u_j^n (i.e., $u_j^n = u(j\Delta x, n\Delta t)$), and the equation evaluated at (x,t^n) is

$$\left(\frac{\partial u}{\partial t}\right)_j^n + f_j^n = 0 \tag{2.4.6}$$

Using a simple forward difference for the time derivative and a central difference for the space derivative, the discretized formulation takes the form

$$u_j^{n+1} = u_j^n - \Delta t\, f\left[\left(u_{j+1}^n - u_{j-1}^n\right)/2\Delta x, u_j^n, x_j, t^n\right] \tag{2.4.7}$$

Given the solution at time level n for $j=1,2,\ldots,n$, the explicit formulation can be used to evaluate directly the solution at time level n+1. As a specific example, the transport equation

$$\frac{\partial u}{\partial t} + a\frac{\partial u}{\partial x} = 0 \tag{2.4.8}$$

where "a" is a constant, can be expressed as the explicit formulation

$$u_j^{n+1} = u_j^n - a\Delta t\left(\frac{u_{j+1}^n - u_{j-1}^n}{2\Delta x}\right) \qquad (2.4.9)$$

A formulation of this type indicates that time can be used to progress the numerical solution to steady state and is often referred to as the method of false transients. In such cases, Eq. (2.4.9) indicates that steady state can be achieved by choosing the time step Δt to be large. However, if this were true, it would appear that Δt could be chosen to achieve steady state in one step. This is clearly unrealistic and can be confirmed by investigating the stability of the numerical formulation.

Stability Analysis

Two main techniques are commonly used to assess the stability of numerical formulations. The first is called the Von Neumann or Fourier method, and the second is called the matrix method.[1]

The Fourier method expresses an initial distribution (line) of errors in terms of a finite Fourier series and considers the growth or decay of the errors. It proves convenient to express the errors in terms of a complex potential form

$$A_n e^{i\beta ph} \qquad (2.4.10)$$

where $i = \sqrt{-1}$, [0,1] is the interval throughout which the function is defined, and h is the grid spacing. Denote the errors at the mesh points along t=0, between x=0 and Nh, by $E(ph) = E_p$, p=0,1,2,....,N. Then the (N+1) equations

$$E_p = \sum_{n=0}^{N} A_n e^{i\beta ph}, \quad (p = 0, 1, \ldots, N)$$

are sufficient to determine the (N+1) unknowns $A_0, A_1, A_2, \ldots, A_N$. Assuming the finite difference equations to be linear, it is necessary to only consider the propagation of a single term, such as $e^{i\beta ph}$. To investigate the propagation of this error as t (defined as t=qk) increases, it is necessary to find a solution of the finite difference equation which reduces to $e^{i\beta ph}$ when t=0. Assume

$$E_{p,q} = e^{i\beta x}e^{\alpha t} = e^{i\beta ph}e^{\alpha qk} = e^{i\beta ph}\xi^q$$

where $\xi = e^{\alpha k}$ and α, in general, is a complex constant. This reduces to $e^{i\beta ph}$ when q=0. The error will not increase as t increases provided that

$$|\xi| \leq 1$$

Hence, the stability of a scheme can be investigated by applying a Fourier mode of the form

$$a_k = \xi^n e^{i\beta x} \qquad (2.4.11)$$

to the discretized equations; the scheme is stable provided $a_k(t)$ does not increase in time for all k.

Applying these ideas to the simple example of the transport equation by substituting Eq. (2.4.11) in Eq. (2.4.9) leads to

$$a_k^{n+1}e^{i\beta x} = a_k^n e^{i\beta x} - \frac{C}{2}\left[a_k^n e^{i\beta(x+\Delta x)} - a_k^n e^{i\beta(x-\Delta x)}\right]$$

which reduces to

$$a_k^{n+1} = a_k^n - \frac{C}{2}[2i\sin(\beta\Delta x)]a^n$$

and

$$a_k^{n+1} = \{1 - iC\sin(\beta\Delta x)\}a_k^n$$

with $C = \frac{a\Delta t}{\Delta x}$. The scheme is stable if

$$|1 - iC\sin(\beta\Delta x)| \leq 1$$

If the amplification factor g, defined as $g = \frac{a_k^{n+1}}{a_k^n}$, is complex, then $|g|^2 = gg^*$, with g^* as the complex conjugate. Hence, the stability condition is

$$1 + C^2\sin^2(\beta\Delta x) \leq 1$$

which is not satisfied for any Δx, $\Delta t > 0$; thus, the simple scheme is unstable and so not of much practical use. In practice the numerical errors produced would grow unboundedly and the numerical answer after a few time steps would be meaningless. It is clear from this simple example that it is not possible to formulate arbitrary explicit schemes.

In the above analysis no account has been taken of the effect on stability of the boundary conditions. The Fourier method is not able to take such effects into account and for such analysis the alternative matrix method of stability is required.[1]

Lax Schemes of First Order Accuracy

It is clear from stability analysis that the method of forward difference in time and central difference in space is unconditionally unstable. To alleviate this problem, Lax[2] proposed a modification to give a stable explicit numerical procedure. As an example consider the following form for the discretized transport Eq. (2.4.8):

$$\frac{u_j^{n+1} - \frac{1}{2}(u_{j+1}^n + u_{j-1}^n)}{\Delta t} + a\frac{u_{j+1}^n - u_{j-1}^n}{2\Delta x} = 0 \qquad (2.4.12)$$

This is similar to the unstable Euler method (2.4.9), except that u_j^n has been replaced by $\frac{1}{2}(u_{j+1}^n + u_{j-1}^n)$.

A Fourier stability analysis of this formulation reveals that

$$g = \cos(\beta\Delta x) - iC\sin(\beta\Delta x)$$

where $C = \frac{a\Delta t}{\Delta x}$. For stability, $|g| \leq 1$, hence,

$$|g|^2 = 1 + \{C^2 - 1\}\sin^2(\beta\Delta x)$$

and so the scheme is stable for $C \leq 1$. C is referred to as the Courant number and for a given constant speed "a" and mesh spacing Δx, it determines the Δt at which the numerical formulation can be stepped in time to steady state.

It is well known why the Lax's scheme is stable while the forward time, central space scheme is unconditionally unstable. The Lax formulation can be written as

$$\frac{u_j^{n+1} - u_j^n - \frac{1}{2}(u_{j+1}^n - 2u_j^n + u_{j-1}^n)}{\Delta t} + a\frac{u_{j+1}^n - u_{j-1}^n}{2\Delta x} = 0$$

or

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} + a\frac{u_{j+1}^n - u_{j-1}^n}{2\Delta x} = \frac{\Delta x^2}{2\Delta t}\frac{(u_{j+1}^n - 2u_j^n + u_{j-1}^n)}{\Delta x^2}$$

which shows that the averaging of the term u_j^n leads to the introduction of a term which is a central difference of a second space derivative. In other words Lax's scheme can be seen as an Euler scheme applied to a modified equation

$$\frac{\partial u}{\partial t} + a\frac{\partial u}{\partial t} = \frac{\Delta x^2}{2\Delta t}\frac{\partial^2 u}{\partial x^2} \qquad (2.4.13)$$

where the right hand side is a term which represents an added diffusion-like term with a grid dependent coefficient. It is the right hand side which provides the stability. A truncation error analysis reveals that the scheme, for a fixed Courant number, is first order accurate in space and time.

The Lax scheme can be applied to the system of Euler equations in two dimensions

$$\frac{\partial \mathbf{w}}{\partial t} + \frac{\partial \mathbf{f}}{\partial x} + \frac{\partial \mathbf{g}}{\partial y} = 0 \qquad (2.4.14)$$

to give, for a structured grid,

$$\mathbf{w}_{ij}^{n+1} = \frac{1}{4}(\mathbf{w}_{i+1,j}^n + \mathbf{w}_{i-1,j}^n + \mathbf{w}_{i,j+1}^n + \mathbf{w}_{i,j-1}^n) - \frac{\Delta t}{2\Delta x}(\mathbf{f}_{i+1,j}^n - \mathbf{f}_{i-1,j}^n) - \frac{\Delta t}{2\Delta y}(\mathbf{g}_{i,j+1}^n - \mathbf{g}_{i,j-1}^n) \qquad (2.4.15)$$

Schemes of this type together with many variants are classed as Lax or Lax-Friedrichs schemes. They have played a major role in the development of numerical schemes. However, they are not generally applied now because of their lack of accuracy.

Lax-Wendroff Schemes of Second Order Accuracy

Schemes with second order accuracy are generally classed as Lax-Wendroff schemes of which today there are many variations on that originally proposed.[3]

The major criticism of the Lax scheme is its first order accuracy. The Lax-Wendroff class of techniques uses central space representation but achieves second order accuracy. Again consider the first order equation

$$\frac{\partial u}{\partial t} + \frac{\partial f}{\partial x} = 0$$

A Taylor series expansion in time shows

$$u^{n+1} = u^n + \Delta t\, u_t + \frac{\Delta t^2}{2}u_{tt} + \frac{\Delta t^3}{6}u_{ttt} \ldots\ldots$$

which, using the original partial differential equation, can be expressed as

$$u^{n+1} = u^n - \Delta t\frac{\partial f}{\partial x} + \frac{\Delta t^2}{2}\frac{\partial}{\partial x}\left(A\frac{\partial f}{\partial x}\right) + O(\Delta t^3)$$

where the Jacobian $A \equiv \frac{\partial f}{\partial u}$. The discretized form of this equation gives

$$u_i^{n+1} = u_i^n - \frac{\Delta t}{2\Delta x}(f_{i+1}^n - f_{i-1}^n) + \frac{1}{2}\left(\frac{\Delta t}{\Delta x}\right)^2\left[A_{i+\frac{1}{2}}^n(f_{i+1}^n - f_i^n) - A_{i-\frac{1}{2}}^n(f_i^n - f_{i-1}^n)\right] \qquad (2.4.16)$$

with either

$$A_{i+\frac{1}{2}} = A\left(u_{i+\frac{1}{2}}\right)$$

or

$$A_{i+\frac{1}{2}} = \frac{1}{2}(A_i + A_{i+1})$$

This scheme requires the evaluation of the Jacobian matrix A, which can be costly. To alleviate this, several two-step schemes have been developed.

One such scheme, which has proved to be very popular, was devised in 1969 by MacCormack.[4,5] The scheme is a two-step predictor-corrector method which for the equation

$$\frac{\partial u}{\partial t} + \frac{\partial f}{\partial x} = 0$$

takes the form

$$u_i^* = u_i^n - \frac{\Delta t}{\Delta x}(f_i^n - f_{i-1}^n)$$
$$u_i^{n+1} = \frac{1}{2}(u_i^n + u_i^*) - \frac{\Delta t}{\Delta x}(f_{i+1}^* - f_i^*) \qquad (2.4.17)$$

as illustrated in Fig. 2.4.1.

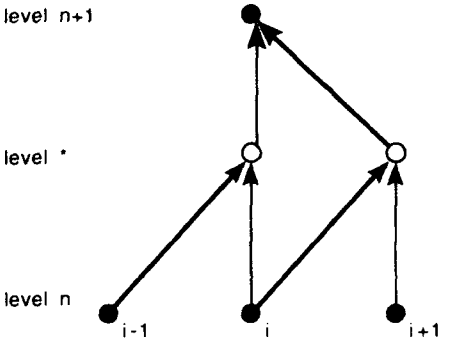


Figure 2.4.1 Information used in the two-step MacCormack scheme.

The scheme is second order accurate in both space and time. The scheme readily extends to both two and three dimensions; for example, in two dimensions, for the equation

$$\frac{\partial u}{\partial t} + \frac{\partial f}{\partial x} + \frac{\partial g}{\partial y} = 0$$

the scheme is

$$u_{ij}^* = u_{ij}^n - \frac{\Delta t}{\Delta x}(f_{ij}^n - f_{i-1,j}^n) - \frac{\Delta t}{\Delta y}(g_{ij}^n - g_{i,j-1}^n)$$
$$u_{ij}^{n+1} = \frac{1}{2}(u_{ij}^n + u_{ij}^*) - \frac{\Delta t}{2\Delta x}(f_{i+1,j}^* - f_{ij}^*) - \frac{\Delta t}{2\Delta y}(g_{i,j+1}^* - g_{ij}^*) \qquad (2.4.18)$$

A technique which is often used in the representation of general schemes is operator splitting. The one-dimensional predictor-corrector sequence in Eq. (2.4.17) is represented by

$$u^{n+1} = L_x \Delta t\, u^n \qquad (2.4.19)$$

Combining i and j operators then gives

$$u^{n+1} = L_i(\Delta t)L_j(\Delta t)u^n \qquad (2.4.20)$$

which is a representation of a 2D scheme. However, this unsymmetrical sequence is only first order accurate. The MacCormack scheme results from a symmetric split sequence of the form

$$u^{n+1} = L_i\left(\frac{\Delta t}{2}\right)L_j(\Delta t)L_i\left(\frac{\Delta t}{2}\right)u^n \qquad (2.4.21)$$

In this form, it is clear how such a scheme could be used to advantage if for some flows the time step is limited by a particular direction, either Δx or Δy. In such cases, a sequence could be devised which takes more steps in one of the directions so as to reduce the computational cost.

Time Stepping Schemes

A common practice for the solution of the time-dependent Euler equations is to first discretize the equations in space while ignoring the time derivative. This semidiscrete procedure then results in a system of ordinary differential equations which must be integrated in time. For steady state problems, the equations are integrated in time using any appropriate integration routine and flexibility is present for convergence acceleration techniques (such as local time stepping) to be used. Such schemes are independent of the time step Δt at convergence to the steady state.

Time stepping procedures can be based upon the schemes available extensively in the literature for the solution of ordinary differential equations. Single-step methods can be used based upon Euler and Taylor series methods. Consider the first-order scalar differential equation

$$\frac{dy}{dx} = f(x, y(x)) \qquad (2.4.22)$$

with the initial conditions

$$y(x_0) = y_0$$

The simplest numerical method for treating this initial value problem consists of approximating the solution curve y(x) by its tangent. With the step size h and the corresponding equidistant support abscissae

$$x_k = x_0 + kh \quad (k = 1, 2, \ldots\ldots)$$

an approximation y_k to the exact solution can be obtained by the general formula

$$y_{k+1} = y_k + hf(x_k, y_k) \qquad (2.4.23)$$

At each point (x_k, y_k), this method (named after Euler) uses the slope of the directional field that is defined by the given differential equation to determine the next approximation y_{k+1}. This procedure is obviously quite coarse and its accuracy is dependent upon the size of h. However, the procedure represents the simplest member of a one step method which uses only the known approximate value y_k at the support abscissa x_k to compute the approximation y_{k+1} at $x_{k+1} = x_k + h$.

An alternative procedure is to note that in a small neighborhood of the initial point (x_0, y_0), a better approximation of the desired solution y(x) can be obtained by means of the Taylor series with a remainder term

$$y(x) = y(x_0) + \frac{(x - x_0)}{1!}y'(x_0) + \frac{(x - x_0)^2}{2!}y''(x_0) + \ldots \frac{(x - x_0)^p}{p!}y^p(x_0) + R_{p+1}$$

If the remainder term R_{p+1} is neglected, the approximate value of y_{k+1} for the step size $h=(x_{k+1}-x_k)$ is given by the general formula

$$y_{k+1} = y_k + \frac{h}{1!}y'_k(k) + \frac{h^2}{2!}y''_k + \ldots + \frac{h^p}{p!}y_k^{(p)}$$

Here, $y_k^{(m)}$ denotes the value of the mth derivative at the point (x_k, y_k). This procedure requires the derivatives of the function y(x) and this, for general cases, can prove inaccurate or unwieldy to compute. However, with a slightly different formulation the method can be used effectively.[7]

Various methods exist for improving these simple formulations. Since Euler's method is of order one, an extrapolation can be applied. Assuming that two integrations have been performed, the first with step h and the second with step h/2 and then using the principles of Richardson extrapolation, an improved Euler method of the form

$$\begin{aligned} k_1 &= f(x_k, y_k) \\ k_2 &= f\left(x_k + \frac{h}{2}, y_k + \frac{1}{2}hk_1\right) \\ y_{k+1} &= y_k + hk_2 \end{aligned} \qquad (2.4.24)$$

can be used. This is a second order formulation; a single step requires the evaluation of the function f(x,y) at two different pairs of values.

Other single step methods for the solution of the differential equations can be obtained by the use of standard definite integration methods. The trapezoidal method can be invoked to derive the formulation

$$y_{k+1} = y_k + \frac{h}{2}(f(x_k, y_k) + f(x_{k+1}, y_{k+1})) \qquad (2.4.25)$$

For a general nonlinear equation this represents an implicit equation for the unknown y_{k+1}.

In practice such schemes can be implemented using a sequence of successive approximations to y_{k+1}, denoted as $y_{k+1}^{(p)}$. One such procedure, known as Heun's method, can be written

$$\begin{aligned} y_{k+1}^{(p)} &= y_k + hf(x_k, y_k) \\ y_{k+1} &= y_k + \frac{h}{2}\left(f(x_k, y_k) + f\left(x_{k+1}, y_{k+1}^{(p)}\right)\right) \end{aligned} \qquad (2.4.26)$$

The explicit first order Euler method (2.4.26a) is used to determine a predictor value $y_{k+1}^{(p)}$ which is subsequently corrected in Eq. (2.4.26b) by means of the implicit trapezoidal method to obtain y_{k+1}. This is called a predictor-corrector method; it is second order accurate and for practical purposes can be rewritten as

$$\begin{aligned} k_1 &= f(x_k, y_k) \\ k_2 &= f(x_k + h, y_k + hk_1) \\ y_{k+1} &= y_k + \frac{h}{2}(k_1 + k_2) \end{aligned} \qquad (2.4.27)$$

The improved Euler method and Heun's method are examples of explicit two-stage second order Runge-Kutta methods.

The formal derivation of Runge-Kutta methods is covered in many standard texts on ordinary differential equations. Here, it is worth quoting the classical fourth-order Runge-Kutta method

$$\begin{aligned} k_1 &= f(x_k, y_k) \\ k_2 &= f\left(x_k + \frac{1}{2}h, y_k + \frac{1}{2}hk_1\right) \\ k_3 &= f\left(x_k + \frac{1}{2}h, y_k + \frac{1}{2}hk_2\right) \\ k_4 &= f(x_k + h, y_k + hk_3) \\ y_{k+1} &= y_k + \frac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4) \end{aligned} \tag{2.4.28}$$

It is clear from these formulations that to compute y_{k+1} they require information based upon the previous approximation point (x_k, y_k). In contrast, an alternative set of methods is to use a multistep approach which uses information at previous support abscissae x_{k-1}, x_{k-2}, x_{k-3}...x_{k-m}. These methods are referred to as linear multistep methods, and they can be either explicit or implicit in time. A classical example is the method of Adams-Bashforth which can be expressed as

$$y_{k+1} = y_k + \frac{h}{24}(55f_k - 59f_{k-1} + 37f_{k-2} - 9f_{k-3}) \tag{2.4.29}$$

This formulation, which assumes equal spacing h, is an explicit, linear four-step method. It requires only one function evaluation at each step, but requires previous values f_k, f_{k-1}, f_{k-2}, f_{k-3}. Other variants of this multistep method exist, as well as generalized formulations.[6]

It is clear from this discussion that multistage, two level schemes of the Runge Kutta type have the advantage that they do not require any special starting procedures, in contrast to the multistep methods of Adams-Bashforth type. In addition, extra stages in the multistage methods can be used to improve accuracy or to extend the stability region. Jameson et al.[7-9] have used Runge-Kutta methods to great advantage for the solution of the Euler equations. In general, the selection of a particular temporal integration scheme is closely tied to the type of discretization selected for the spatial derivatives.

The extension of these ideas discussed for the one-dimensional model problem to the governing flow equations is straightforward. The discretization of spacial operators in the equations leads to a system of coupled ordinary differential equations of the form

$$\frac{d\mathbf{w}}{dt} + \mathbf{R}(\mathbf{w}) = 0 \tag{2.4.30}$$

where $\mathbf{R}(\mathbf{w})$ is the vector of the residuals. Let $\mathbf{w}^n$ be the numerical result after n time steps; the formulation to advance the solution to time level n+1 is

$$\begin{aligned} \mathbf{w}^{(0)} &= \mathbf{w}^n \\ \mathbf{w}^{(1)} &= \mathbf{w}^{(0)} - \alpha_1 \Delta t \mathbf{R}^{(0)} \\ &\cdots\cdots\cdots \\ \mathbf{w}^{(m-1)} &= \mathbf{w}^{(0)} - \alpha_{m-1} \Delta t \mathbf{R}^{(m-2)} \\ \mathbf{w}^{(m)} &= \mathbf{w}^{(0)} - \Delta t \mathbf{R}^{(m-1)} \\ \mathbf{w}^{n+1} &= \mathbf{w}^{(m)} \end{aligned} \tag{2.4.31}$$

The residual in the q+1 stage is evaluated as

$$\mathbf{R}^{(q)} = \sum_{r=0}^{q} \beta_{qr} \mathbf{R}\left(\mathbf{w}^{(r)}\right) \tag{2.4.32}$$

where

$$\sum_{r=0}^{q} \beta_{qr} = 1$$

Typically, three-, four-, or five-stage schemes have been extensively used.

The stability properties of the multistage method have been used to advantage in the solution of the Euler equations. Consider a time stepping scheme for

$$\Delta t \frac{du}{dt} = \lambda u$$

The stability region is the region of the complex plane for which the scheme is stable in terms of the time step multiplied by the eigenvalues of A; the time step is selected so that the eigenvalues of the A matrix lie within the stability region. For a multistage scheme,

$$\begin{aligned} u^{(1)} &= u^{(0)} + \alpha_1 \lambda u^{(0)} \\ u^{(2)} &= u^{(0)} + \alpha_2 \lambda u^{(1)} \\ u^{(3)} &= u^{(0)} + \alpha_3 \lambda u^{(2)} \\ &\cdots\cdots\cdots \end{aligned}$$

Thus, for a complete step

$$u^n = g u^n$$

where the amplification factor g is a polynomial in A:

$$g = 1 + \beta_1 \lambda + \beta_2 \lambda^2 + \ldots$$

and $\beta_1 = 1$, $\beta_2 = \alpha_{k-1}$, $\beta_3 = \alpha_{k-1}\alpha_{k-2}$, ... The stability region is the region in which $|g| \le 1$, and we require the Fourier symbol $z(\zeta) = z(\beta \Delta x)$ to lie in the region of stability for all wave numbers $0 < \zeta \le \pi$. Figure 2.4.2 shows the stability region for commonly used Runge-Kutta methods.

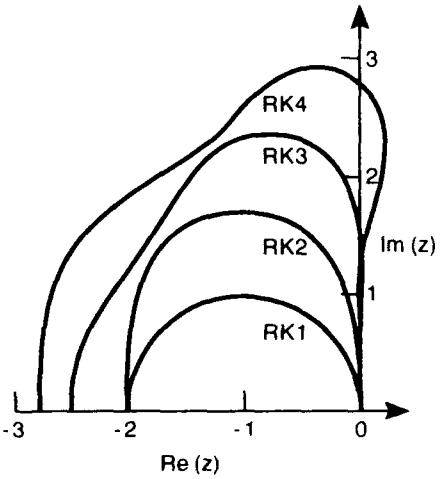


Figure 2.4.2 Stability regions for explicit Runge-Kutta methods.

The equation

$$u_t + u_x + m\Delta\xi u_{xxxx} = 0$$

is more typical of an appropriate model equation for the Euler equations, where dissipation is added explicitly to the convective terms. In this case, modifications to the stability plots can be obtained with single or multiple evaluations of the dissipationlike term within the multistage procedure. The variety of possibilities have been considered by Jameson.[10]

2.4.2 Implicit Schemes

In the discussion that follows, the system of ordinary differential equations to be solved is written as

$$\frac{d}{dt}\left(\frac{\mathbf{w}}{J}\right) + \mathbf{R}(\mathbf{w}) = 0 \tag{2.4.33}$$

where $\mathbf{R}(\mathbf{w})$ is the discrete representation of the spatial derivative terms evaluated as a function of the solution vector $\mathbf{w}$. The prototype implicit algorithm considered is the backward time differencing equation, given as

$$\frac{\Delta\mathbf{w}^n}{J\,\Delta t} + \mathbf{R}^{n+1} = 0 \tag{2.4.34}$$

where Δt is the time step, $\mathbf{R}^{n+1}$ is the discrete residual evaluated at time level $(n+1)\,\Delta t$, and $\Delta\mathbf{w}^n \equiv \mathbf{w}^{n+1} - \mathbf{w}^n$ is the change in the dependent variables over a time step. The scheme is first order accurate in time. The Trapezoidal scheme can be written as

$$\frac{\Delta\mathbf{w}^n}{J\,\Delta t} + \frac{1}{2}\left[\mathbf{R}^n + \mathbf{R}^{n+1}\right] = 0 \tag{2.4.35}$$

and the three–point backward-time scheme can be written as

$$\frac{1}{J\,\Delta t}\left[\frac{3}{2}\,\Delta\mathbf{w}^n - \frac{1}{2}\,\Delta\mathbf{w}^{n-1}\right] + \mathbf{R}^{n+1} = 0 \tag{2.4.36}$$

both of which are second order accurate in time. The three schemes above can be considered as examples from the well-known class of linear multistep methods developed for solving ordinary differential equations. The stability of such methods can be determined from an analysis of the eigenvalues of the coefficient matrix arising from linearization of the nonlinear terms. For discrete solutions to the Euler equations, these eigenvalues generally lie in the left half of the complex plane. A method without any time step stability limit in such a case is referred to as an A-stable method. It is known from a theorem of Dahlquist that:

(1) The order of an A-stable method cannot exceed two.

(2) The second-order A-stable scheme with lowest truncation error is the Trapezoidal scheme.

Generally, the three–point backward-time scheme is preferred for second order accuracy since the Trapezoidal method, also known as Crank-Nicholson, is susceptible to an odd-even decoupling in time of the highest frequencies in the solution.

Either Eq. (2.4.34) or (2.4.36) represents a nonlinear system of equations to be solved at each time step and can be written generically as

$$\hat{\mathbf{R}} = 0 \tag{2.4.37}$$

where the hatted notation denotes that the residual contains both temporal and spatial discretization terms. Applying Newton's method for the root of a nonlinear system of equations gives a linear system to be solved iteratively,

$$\hat{\mathbf{R}}^{i+1} \approx \hat{\mathbf{R}}^i + \left[\frac{\partial\hat{\mathbf{R}}}{\partial\mathbf{w}}\right]^i \left(\mathbf{w}^{i+1} - \mathbf{w}^i\right) \equiv 0, \qquad i = 1, 2, 3, \ldots \tag{2.4.38}$$

The linearization is about an estimate $\mathbf{w}^i$, which can be taken initially as $\mathbf{w}^n$, and the solution converges to the solution at the new time level $\mathbf{w}^{n+1}$. The requirement to solve a linear equation arises from the linearization of the nonlinear spatial residual terms at the new time level. In that respect, the treatment of second order accuracy in time is similar to that of the first order scheme, since the additional terms involved are all evaluated at time levels which have already been computed. Hence, restricting the discussion below to the first-order backward time scheme, the linear system is written as

$$\left[\mathbf{I} + h\,\frac{\partial\mathbf{R}}{\partial\mathbf{w}}\right]^i \left(\mathbf{w}^{i+1} - \mathbf{w}^i\right) = -\left[\mathbf{w}^i - \mathbf{w}^n + h\,\mathbf{R}\left(\mathbf{w}^i\right)\right], \qquad i = 1, 2, 3, \ldots \tag{2.4.39}$$

where $h = J\Delta t$. For each iteration, Eq. (2.4.39) requires the solution of a block-banded linear system of equations with the property that quadratic convergence can be attained at each iteration if the approximation is sufficiently near the root of the equation. For large time steps, Newton's method is recovered for the solution of the steady-state residual equation.

A general block-matrix equation solved with Gauss elimination requires

$$O(N^3 m^3)\ operations \tag{2.4.40}$$

where N is the total number of equations and m is the block size. A general block-banded matrix equation can be solved in

$$O(Nb^2 m^3)\ operations;\ b = max(p, q) \tag{2.4.41}$$

where p is the number of nonzero off-diagonals in the matrix at or above the main diagonal and q is the number at or below the main diagonal of the matrix to be solved.

The computational work for typical structured-grid solvers can be estimated assuming an implicit computational stencil which spans three points in each coordinate direction. For a three-dimensional ordering of the unknowns by generalized coordinate directions,

$$N = J \cdot K \cdot L$$
$$p = q = J \cdot K$$

where J, K, L is the number of points in each of the three coordinate directions, respectively. For a two-dimensional case with ordering by rows,

$$N = J \cdot K$$
$$p = q = J$$

Hence, the computational work scales as

$$O\left(N^{7/3} m^3\right)\ operations: 3-D \tag{2.4.42}$$

$$O(N^2 m^3)\ operations: 2-D \tag{2.4.43}$$

assuming equal number of points in each coordinate direction. In two dimensions, the relative bandwidth is smaller which results in a smaller operation count; since both N and m are usually smaller in this case, direct solutions of the linear system are possible with available computers,[11,12] at least for steady-state solutions.

Because of the number of operations and storage involved in a direct solution of the linear system at each iteration, the complexity which can arise in linearizing the equations exactly, and the realization that quadratic convergence is only obtained when the approximation is near to the exact solution, the method as outlined above is rarely used in application.[12] A number of approximate techniques have been devised. Approximate linearizations, denoted with double tilde overscripts, can be used on the left side of the equation (1) to reduce the bandwidth of the linear system or (2) to reduce the complexity associated with an exact linearization, as

$$\left[\mathbf{I} + h\,\frac{\partial \tilde{\tilde{\mathbf{R}}}}{\partial \mathbf{w}}\right]^{i}\left(\mathbf{w}^{i+1} - \mathbf{w}^{i}\right) = -\left[\mathbf{w}^{i} - \mathbf{w}^{n} + h\,\mathbf{R}\left(\mathbf{w}^{i}\right)\right] \qquad i = 1, 2, 3, \ldots \tag{2.4.44}$$

Since the equation is cast in delta form,[13] the nonlinear equation on the right-hand side will be satisfied as long as the sequence of iterates converge. With an approximate linearization, the property of quadratic convergence at a minimum is lost. Furthermore, all approximate linearizations are not stable, even if the linear system is solved exactly.[14,15] For example, Jesperson and Pulliam[14] show a one-dimensional stability analysis corresponding to steady-state convergence of a second-order accurate scheme; an exact linearization with a first-order implicit stencil was unconditionally stable. However, with a second-order implicit stencil but an approximate Jacobian (linearization terms), stability was only conditional.

Another approach is to determine a solution to the linear system but accept the first iteration as the solution at the new time level, as a so-called single-step scheme,[13,16] i.e.,

$$\left[\mathbf{I} + h\,\frac{\partial \mathbf{R}}{\partial \mathbf{w}}\right]^{n}\left(\mathbf{w}^{n+1} - \mathbf{w}^{n}\right) = -\left[h\,\mathbf{R}(\mathbf{w}^{n})\right] \tag{2.4.45}$$

The equation is now of the standard form $\mathbf{A}\mathbf{x} = \mathbf{b}$, where $\mathbf{x} \equiv \Delta\mathbf{w}$, and a number of iterative methods adapted from the study of linear algebraic equations can be applied to solve the linear system of equations. Note that the convergence of the solution to the linear system depends only on the coefficient matrix. The use of large time steps and the retention of the A-stable properties of the implicit integration scheme can be attained if the linear system is solved to near completion. However, in most cases, the linear equation need not be solved exactly, since for steady flows the ultimate objective is the solution to a nonlinear equation. Likewise, for unsteady simulations, it is necessary only to solve the equation to a tolerance below that of the truncation and linearization errors associated with the single-step approximation.

The two approximations can be combined, as an approximately linearized implicit scheme, in which the linear equation is hopefully much easier to solve, as

$$\left[\mathbf{I} + h\,\frac{\partial \tilde{\tilde{\mathbf{R}}}}{\partial \mathbf{w}}\right]^{n}\left(\mathbf{w}^{n+1} - \mathbf{w}^{n}\right) = -\left[h\,\mathbf{R}(\mathbf{w}^{n})\right] \tag{2.4.46}$$

A number of schemes are in current use which are based on this model. For example, since this coefficient matrix depends only on the left-side spatial-difference approximation of Eq. (2.4.45), first-order upwind differencing can often be used to ensure block-diagonal dominance of the coefficient matrix and relaxation methods can be used effectively. Also, the approximate factorization methods discussed subsequently are a special case of the algorithms described by Eq. (2.4.46).

Direct Methods

Direct methods can be used to solve the system of linear equations associated with the implicit schemes above. Direct methods are distinct from iterative solvers in that they solve the system of equations in a finite and predetermined amount of work. Solutions to the linear system using direct methods are exact if infinite precision arithmetic is used. On computers, direct solvers will generate solutions that are as accurate as the arithmetic used to generate them.

The coefficient matrices resulting from Euler solvers are generally sparse (i.e., composed of few nonzero and many zero coefficients). Linear systems resulting from structured grid Euler solvers will have an ordered structure of nonzero coefficients. For example, a two-dimensional first-order upwind scheme results in a pentadiagonal matrix composed of two "diagonal entries" clustered around the central diagonal and a diagonal entry farther out on either side. Adding more dimensions and/or higher order accuracy leads to a matrix composed of more diagonal entries. This form of sparseness is known as structural sparseness since it is known a priori which coefficient elements are zero and which are nonzero. The use of unstructured grid Euler solvers leads to implicit matrices with nonzero coefficients located on the diagonal entry and randomly located off the diagonal entry.

Specialized versions of Gaussian elimination are used for sparse linear systems to minimize storage costs and reduce operation counts.[18-20] Banded-matrix direct solvers are perhaps the most common approach to reducing the storage and operation count of a full Gaussian elimination. Banded solvers store diagonal entries of the matrix as vectors and, hence, store all coefficient elements of the matrix out to the last diagonal entry that has a nonzero coefficient in it. In large part, most work on banded direct solvers has been done in the structural finite-element field. Consequently, most banded solvers are specialized for symmetric positive-definite matrices.

Matrices that possess general, rather than structural, sparseness have nonzero coefficients randomly located in the array. Consequently, very few nonzero coefficients exist next to the maximum bandwidth of the matrix. Skyline solvers take advantage of this fact by only storing row or column vectors from the diagonal to the last nonzero coefficient in each row (lower triangular) or column (upper triangular), respectively. Skyline solvers are also known as envelope, profile, or variable bandwidth solvers.

The most common approach to solving general sparse matrices has been to renumber the grid to minimize the bandwidth or minimize the numerical fill-in during the solution process. Bandwidth minimization will make both band solvers and skyline solvers much more efficient for any linear problem. Several methods have been proposed to minimize the bandwidth of sparse linear systems. The most successful and best known of

the bandwidth minimization algorithms is the Cuthill-McKee[18] ordering algorithm. This algorithm makes extensive use of graph theory as do most reordering algorithms. For each node (equation), a list of adjacent nodes can be generated from the graph of the matrix. If v is the node number, then $Adj(v)$ is a list of the nodes immediately adjacent to v. The algorithm takes a list of nodes and the associated list of adjacent nodes and generates a new list of nodes. The old and new list of nodes acts as a translation table for the matrix so that the graph of the matrix may be renumbered. The matrix associated with the new graph of the nodes will often have a dramatically reduced bandwidth.

Matrix dissection,[18-20] or nested dissection, is a method to renumber the graph of a matrix. The bandwidth of the matrix is not reduced but the numerical fill-in that occurs during the matrix factorization is greatly reduced. The minimum degree method also reduces fill-in through a minimization process and has proved useful for unstructured grids.[20]

The principal advantage of the direct solvers is their robustness, since steady-state solutions can be generated in cases where iterative methods fail to converge and there are fewer parameters to adjust for improved convergence.[11,12,20,21] The rapid convergence of the scheme allows solutions to be obtained in four to five iterations; because of the increased operation count of the direct solution, the method is generally only invoked after an approximate solution is generated with an iterative solver.

Approximate Factorization Methods

Within the framework of approximate factorization (AF) methods, implicit schemes which factor spatially the unsplit matrix equation into a sequence of simpler matrix equations are known as alternating direction implicit (ADI) schemes[13,16] and have been widely used. In addition to the classical spatially factored scheme, a number of alternative schemes are possible by factoring the implicit operator according to the eigenvalues of the split Jacobian matrices and using type-dependent differencing.[22-24] These alternate factorizations can be used to split the full operator into a lower (L) and an upper (U) factor independent of the number of spatial dimensions of the problem, thereby increasing the allowable time step based on stability considerations and/or a decrease in the number of operations. These LU schemes can be used to converge discretizations corresponding to either central or upwind schemes (e.g., Rieger and Jameson[24]).

For the compressible Euler and Navier-Stokes equations, Beam and Warming[13] and Briley and MacDonald[16] laid the foundations of the current ADI algorithms which are generalizations of the alternating-direction implicit algorithms developed in the 1950's for solving parabolic equations. The ADI algorithms approximate the left-hand implicit matrix of Eq. (2.4.39) as a product of one-dimensional matrices, solved in a series of sweeps through the mesh. The nonlinear implicit scheme for first-order backward-time differencing can be written

$$\left[\mathbf{I} + h\,\delta_\xi\left(\frac{\partial \hat{\mathbf{F}}}{\partial \mathbf{w}}\right)\right]^n \Delta\mathbf{w}^{**} = \tilde{\mathbf{R}}^n \tag{2.4.47}$$

$$\left[\mathbf{I} + h\,\delta_\eta\left(\frac{\partial \hat{\mathbf{G}}}{\partial \mathbf{w}}\right)\right]^n \Delta\mathbf{w}^{*} = \Delta\mathbf{w}^{**} \tag{2.4.48}$$

$$\left[\mathbf{I} + h\,\delta_\zeta\left(\frac{\partial \hat{\mathbf{H}}}{\partial \mathbf{w}}\right)\right]^n \Delta\mathbf{w}^{n} = \Delta\mathbf{w}^{*} \tag{2.4.49}$$

$$\mathbf{w}^{n+1} = \mathbf{w}^n + \Delta\mathbf{w}^n \tag{2.4.50}$$

where $\tilde{\mathbf{R}}^n$ denotes the right side of Eq. (2.4.39) and δ is a spatial difference operator. Each sweep requires the solution of a block tridiagonal or pentadiagonal matrix equation. The computational molecule for a three-point spatial differencing is shown in Fig. 2.4.3(a). Since the solution on a given line in the grid decouples from the other lines on a sweep, the operations can be performed efficiently on either vector-processing or parallel-processing computers by simultaneous solution of the linear system along parallel coordinate lines in the mesh. On the basis that the factorization and linearization errors can be considered small, the algorithm is usually applied as a single-step noniterative scheme

$$\mathbf{L}_\xi\,\mathbf{L}_\eta\,\mathbf{L}_\zeta\,(\mathbf{w}^{n+1} - \mathbf{w}^n) = -[h\,\mathbf{R}(\mathbf{w}^n)] \tag{2.4.51}$$

for both steady and unsteady applications. The spatial factorization is largely independent of the type of spatial differencing (i.e. central or upwind). The algorithm is widely used, in part because the thin-layer form of the viscous terms can be easily accommodated; cross-derivative terms arising from mixed-derivative terms are difficult to treat with the spatially factored algorithm, however, and are generally treated explicitly or lagged in time.

Pulliam and Chaussee[25] developed a diagonalized scheme in order to reduce the number of operations associated with the ADI solution. The similarity transformation of the inviscid Jacobian matrix is used to derive an approximate set of scalar equations on each sweep, as shown for the left-hand side of the ξ sweep:

$$\mathbf{T}_\xi\,[\mathbf{I} + h\,\delta_\xi\mathbf{\Lambda}_\xi]\,\mathbf{T}_\xi^{-1}\,\Delta\mathbf{w}^{**} \tag{2.4.52}$$

where $\mathbf{A} = \partial\hat{\mathbf{F}}/\partial\mathbf{w} \equiv \mathbf{T}_\xi\mathbf{\Lambda}_\xi\mathbf{T}_\xi^{-1}$. The linear system is uncoupled since $\mathbf{\Lambda}_\xi$ is a diagonal matrix of eigenvalues; note that the similarity transformation matrix has been moved outside the differencing operator to achieve the efficiency. The original block-tridiagonal or block-pentadiagonal inversion is replaced by scalar inversions and two matrix multiplications, leading to approximately a factor of two reduction in the overall computational time of the ADI algorithm, with generally no appreciable loss in convergence.

Obayashi et al.[26] have used flux splitting to simplify the matrix inversions further by factoring each sweep into two bidiagonal inversions. Applying type-dependent one-sided differencing to the eigenvalue-split components of the Jacobian matrix as

$$\delta_\xi\mathbf{A} = \delta_\xi^-\mathbf{A}^+ + \delta_\xi^+\mathbf{A}^- \tag{2.4.53}$$

where δ_ξ^- and δ_ξ^+ are backward and forward difference operators, respectively; the left-hand side of the ξ sweep can be represented with the LU-ADI scheme as

$$\mathbf{T}_\xi\,[\mathbf{I} + h\,\delta_\xi^-\mathbf{\Lambda}_\xi^+]\,[\mathbf{I} + h\,\delta_\xi^+\mathbf{\Lambda}_\xi^-]\,\mathbf{T}_\xi^{-1}\,\Delta\mathbf{w}^{**} \tag{2.4.54}$$

To enhance the diagonal dominance of the equations, the implicit equation on each sweep can be factored slightly differently following the work of Lombard et al.[27] as

$$\mathbf{T}_\xi\,[\mathbf{I} + h\,(\delta_\xi^-\mathbf{\Lambda}_\xi^+ - \mathbf{\Lambda}_\xi^+)]\,[\mathbf{I} + h\,(\mathbf{\Lambda}_\xi^+ - \mathbf{\Lambda}_\xi^-)]^{-1}\,[\mathbf{I} + h\,(\delta_\xi^+\mathbf{\Lambda}_\xi^- + \mathbf{\Lambda}_\xi^-)]\,\mathbf{T}_\xi^{-1}\,\Delta\mathbf{w}^{**} \tag{2.4.55}$$

where the diagonal entry of each bidiagonal inversion becomes $\mathbf{D} = \mathbf{I} + h|\mathbf{A}_\xi|$. Termed LDU-ADI,[28] the operator in each direction can be considered a single iteration of a symmetric Gauss-Seidel relaxation sweep, as discussed in greater detail later. The extension of the diagonalized methods to the viscous equations is not so straightforward, since these terms cannot be simultaneously diagonalized with the inviscid terms and are either ignored or represented approximately,[29] as, for example, by approximating the viscous Jacobian with its spectral radius.

The largest deficiency of the spatially factored approach is that the factored operator incurs a splitting error in three dimensions which is proportional to the cube of the time step, $(\Delta t)^3$, and the resulting algorithm is only conditionally stable at best.[25,30] For example, South[31] has presented a proof that the three-factor spatially split AF scheme with central differencing is unconditionally unstable according to a Von Neumann stability analysis for a scalar three-dimensional model, time-dependent hyperbolic equation. In practice, the addition of artificial viscosity can be applied to stabilize the scheme, as the second-order upwind scheme (which can be viewed as a central difference scheme plus dissipation) is conditionally stable. In two dimensions, the splitting error is only $(\Delta t)^2$ and unconditional stability can be attained, although the splitting error causes a general deterioration in convergence rate as Δt tends to infinity. The LU schemes, on the other hand, incur only $(\Delta t)^2$ splitting errors in two and three dimensions and can attain larger stability bounds.

Considering only the Euler equations and factoring the operator according to the positive and negative eigenvalues of the split and type-dependently differenced Jacobian matrices,[22] a two-factor scheme is

$$\left\{\mathbf{I} + h(\delta_\xi^- \mathbf{A}^+ + \delta_\eta \mathbf{B}^- + \delta_\zeta^- \mathbf{C}^+)\right\}^n \cdot \left\{\mathbf{I} + h(\delta_\xi^- \mathbf{A}^+ + \delta_\eta^+ \mathbf{B}^- + \delta_\zeta^+ \mathbf{C}^-)\right\}^n \Delta\mathbf{w}^n = \tilde{\mathbf{R}}^n \quad (2.4.56)$$

Defining $\mathbf{L}^+$ and $\mathbf{L}^-$ as the factors containing the backward and forward difference, respectively, the scheme can be written as

$$\mathbf{L}^+ \mathbf{L}^- \Delta\mathbf{w}^n = \tilde{\mathbf{R}}^n \quad (2.4.57)$$

The computational molecule is shown in Fig. 2.4.3(b), and the operation counts for several schemes are given in Table 2.4.1. The operation count for the eigenvalue-factorization scheme is reduced to 25 percent of that required with the ADI scheme,[30] since only bidiagonal inversions are required on each of the two LU sweeps. Originally thought to be inefficient on vector-processing computers, the operations can, in fact, be vectorized by simultaneous operations along $i+j+k = \text{Constant}$ coordinates lines.[32,33] The algorithm is unconditionally stable in three dimensions. For a scalar wave equation, the eigenvalue-split scheme becomes a direct solver for the unfactored equations. The results from a linearized analysis of the coupled Euler equations at a Mach number of 0.8 for several schemes is shown in Fig. 2.4.4. The results indicate unconditional stability with the two-factor eigenvalue-split scheme;[30] the spatially split scheme shows only conditional stability. However, as is true of factored schemes in general, the optimum damping of the error occurs at relatively small time steps, on the order of a time step of five to ten times that of an explicit scheme.

The extension of the methodology for viscous flows is not obvious; however, the viscous terms can be included by expressing the three-point diffusion operator as the sum of a forward and backward difference and factoring accordingly, although some form of approximation is generally required for the compressible equations. For example, a spectral radius scaling of the viscous Jacobian matrices is readily incorporated, similar to that used for the diagonalized schemes. Often, as in the work of Gatlin and Whitfield,[13] the implicit viscous terms are ignored without an adverse effect on the stability of high Reynolds number viscous flows.

Scheme	Operations per point per factor		Total operations per point
	LU decomposition	Back substitution	
Three-factor spatially-split	550	145	695 × 3 = 2085
2-factor eigenvalue-split	225	45	270 × 2 = 540
2-factor hybrid	600	145	745 × 2 = 1490

Table 2.4.1 Operation counts for solving the implicit approximate factorization schemes.

Relaxation Methods

With the development and use of upwind discretizations for the Euler equations, Chakravarthy[34] and Van Leer and Mulder[35] observed that the linearized implicit equations can be solved efficiently with classical relaxation methods. For example, with first-order upwind differencing, the coefficient matrix on the left side of Eq. (2.4.39) can be constructed to be block-diagonally dominant for any time step and standard relaxation techniques, such as Jacobi or line Gauss-Seidel, for the iterative solution of large linear systems can be used. Also, for supersonic flows, relaxation schemes can be constructed to recover efficient space-marching schemes.[36,37] For higher order spatial differencing, the coefficient matrix is no longer diagonally dominant; it is more difficult to construct efficient schemes, and generally symmetric Gauss-Seidel schemes are required.[38,39]

There are two general approaches to incorporating relaxation techniques for the Navier-Stokes equations. The first approach is to solve the linear system via relaxation to near completion before updating the residual equation. Thus, as mentioned previously, using first-order upwind differencing, the coefficient matrix can often be constructed to be block-diagonally dominant. The second approach is to approximate the left-side matrix of Eq. (2.4.39) as a diagonal (Jacobi) or bidiagonal (Gauss-Seidel) matrix and solve for a sequence of iterations, with a nonlinear evaluation of the residual after every pass through the mesh. At convergence, the nonlinear Eq. (2.4.34) is satisfied. The convergence depends on the spatial differencing of both the left and right sides of Eq. (2.4.39).[38]

The first approach is illustrated below for the difference equations arising from the Euler equations. Considering Eq. (2.4.39) as an equation to be solved at each iteration for $\Delta\mathbf{w}^n$,

a sequence of subiterates $\Delta\mathbf{w}^{m+1}$, $m = 1, 2, 3, \ldots$, is computed. Using a symmetric point Gauss-Seidel procedure, one sweep through the mesh with $\Delta\mathbf{w}^{0+1} = 0$ can be written as below

$$[\mathbf{I} + h(\delta_\xi^- \mathbf{A}^+ + \delta_\eta^- \mathbf{B}^+ + \delta_\zeta^- \mathbf{C}^+ - \mathbf{A}^- - \mathbf{B}^- - \mathbf{C}^-)]\,\Delta\mathbf{w}^{1+1} = \tilde{\mathbf{R}}^1 \quad (2.4.58)$$

The second pass becomes

$$[\mathbf{I} + h(\delta_\xi^+ \mathbf{A}^- + \delta_\eta^+ \mathbf{B}^- + \delta_\zeta^+ \mathbf{C}^- + \mathbf{A}^+ + \mathbf{B}^+ + \mathbf{C}^+)]\,\Delta\mathbf{w}^{2+1} = \tilde{\mathbf{R}}^1 - h(\delta_\xi^- \mathbf{A}^+ + \delta_\eta^- \mathbf{B}^+ + \delta_\zeta^- \mathbf{C}^+ - \mathbf{A}^+ - \mathbf{B}^+ - \mathbf{C}^+)\,\Delta\mathbf{w}^{1+1} \quad (2.4.59)$$

On subsequent passes, Eqs. (2.4.58)–(2.4.59) are solved repeatedly, except Eq. (2.4.58) is modified to use available values of $\Delta\mathbf{w}^{m+1}$ from the previous iteration. If the iterations to the linear system are continued sufficiently far to convergence, Eq. (2.4.39) is classified as a quasi-Newton method, since approximate linearizations of the right-hand-side are generally used to simplify the operations or ensure diagonal dominance. The second pass can be rewritten using Eq. (2.4.58) and the two passes written as below

$$[\mathbf{I} + h(\delta_\xi^- \mathbf{A}^+ + \delta_\eta^- \mathbf{B}^+ + \delta_\zeta^- \mathbf{C}^+ - \mathbf{A}^- - \mathbf{B}^- - \mathbf{C}^-)]\,\Delta\mathbf{w}^{1+1} = \tilde{\mathbf{R}}^1 \quad (2.4.60)$$

$$[\mathbf{I} + h(\delta_\xi^+ \mathbf{A}^- + \delta_\eta^+ \mathbf{B}^- + \delta_\zeta^+ \mathbf{C}^- + \mathbf{A}^+ + \mathbf{B}^+ + \mathbf{C}^+)]\,\Delta\mathbf{w}^{2+1} = \mathbf{D}\,\Delta\mathbf{w}^{1+1} \quad (2.4.61)$$

where $\mathbf{D} \equiv \mathbf{I} + h\,(|\mathbf{A}| + |\mathbf{B}| + |\mathbf{C}|)$ is a 5×5 block-diagonal matrix in three dimensions. Defining $\overline{\mathbf{L}^-}$ and $\overline{\mathbf{L}^+}$ as the diagonally dominant factors containing the backward and forward differences, respectively, the scheme can be written as

$$\overline{\mathbf{L}^+}\,\mathbf{D}^{-1}\,\overline{\mathbf{L}^-}\,\Delta\mathbf{w}^1 = \tilde{\mathbf{R}}^1 \quad (2.4.62)$$

Thus the symmetric Gauss-Seidel, Eqs. (2.4.60)–(2.4.61), can be viewed as an approximate factorization,[32] termed LU-SGS,[40] and consequently considered as a time accurate scheme for unsteady applications. The complete viscous terms can be easily incorporated into the algorithm, and Chakravarthy[41] has shown that the cross-derivative terms can be differenced to enhance the diagonal dominance of the coefficient matrix. Of the schemes considered above, the relaxation scheme is the only one, other than direct inversion, for which the cross-derivative terms can be treated implicitly. Note that the relaxation approach can be incorporated directly into unstructured grid methods since the coefficient matrix can be considered to be a general banded system to be solved iteratively. The factored schemes rely on a regular ordering of the grid and are not applicable to unstructured grid methods.

The second approach, in which nonlinear residual evaluations are used in each sweep of the mesh, can be written as

$$\begin{aligned}\overline{\mathbf{L}^+}\,\Delta\mathbf{w}^* &= \tilde{\mathbf{R}}^n \\ \overline{\mathbf{L}^-}\,\Delta\mathbf{w}^{**} &= \tilde{\mathbf{R}}^* = -(\mathbf{w}^* - \mathbf{w}^n + h\,\mathbf{R}^*)\end{aligned} \quad (2.4.63)$$

corresponding to a forward and a backward sweep. Expanding $\tilde{\mathbf{R}}^*$ in a Taylor series about $\tilde{\mathbf{R}}^n$ and retaining only the first-order terms, the total change over the two sweeps is composed of the sum of the two sweeps as

$$\Delta\mathbf{w}^n = \Delta\mathbf{w}^{**} + \Delta\mathbf{w}^* \quad (2.4.64)$$

Combining the two sweeps into a single step-scheme, the scheme can be written as

$$\overline{\mathbf{L}^-}\,\mathbf{D}^{-1}\,\overline{\mathbf{L}^+}\,\Delta\mathbf{w}^n = \tilde{\mathbf{R}}^n \quad (2.4.65)$$

which is very similar to the symmetric Gauss-Seidel relaxation Eqs. (2.4.60)–(2.4.61).

Eberle et al.[42,53] have developed implicit solvers based on the use of approximate implicit Jacobian matrices and vectorizable relaxation schemes. The schemes are designed so that the time step can be arbitrarily large for steady-state applications or taken as a global constant along with subiterations for time-accurate simulations. A number of different transformations from the usual conservation variables as the *implicit* variables in delta form have been used to improve the robustness of the procedure at higher Mach numbers. Brenneis, Eberle, and Schmatz[45] show the effect of large aspect ratio in reducing or even eliminating the diagonal dominance of upwind schemes in more than one dimension; with the use of alternate variables this problem can be overcome. It is interesting to note that several linear upwind schemes which are unstable for model hyperbolic equations according to the usual Von Neumann stability analysis have been used routinely in applications to the full Euler equations.[50] In the applications, the differencing stencils vary because of the nonlinear limiters used to maintain monotonic results near discontinuities and the coefficients of the linearized equations change from point to point in the grid; both effects are neglected in the Von Neumann analysis.

It is sufficient for diagonal dominance of the matrix equations to split the plus and minus Jacobian matrices into matrices with nonnegative and nonpositive eigenvalues, respectively. The convergence of the quasi-Newton process, however, depends on the form of the linearization; the convergence is improved as the implicit Jacobian matrix approaches that of a true linearization of the residual equation to be satisfied at convergence. A true linearization is often difficult or impractical to attain in practice, as shown for example by Barth[54] using the flux-differencing splitting of Roe. All approximate linearizations are not stable, as discussed earlier. A crude estimate of the Jacobian matrices using the spectral radius (i.e. $\mathbf{A}^\pm = (\mathbf{A} \pm \rho_A \mathbf{I})/2$), results in a strictly diagonal form for the matrix $\mathbf{D} = \mathbf{I} + h\,(\rho_A + \rho_B + \rho_C)\mathbf{I}$ and, consequently, a reduced operation count for the LU-SGS scheme. This simple linearization proved effective in the incompressible viscous flow calculations of Yoon and Kwak;[40] the reduced operation count of the scheme with the approximate Jacobian compensates for the reduction in convergence per iteration.

Hybrid Methods

An alternate LU factorization can be derived by splitting the Jacobian matrices in a single coordinate direction, generally in the direction tangential to the body surface.[55] The algorithm can be written as

$$\begin{aligned}&\left[\mathbf{I} + h(\delta_\xi^- \mathbf{A}^+ + \delta_\eta \mathbf{B})\right]^{-1} \cdot \\ &\left[\mathbf{I} + h(\delta_\xi^+ \mathbf{A}^- + \delta_\zeta \mathbf{C})\right]^{-1} \Delta\mathbf{w}^1 = \tilde{\mathbf{R}}^1\end{aligned} \quad (2.4.66)$$

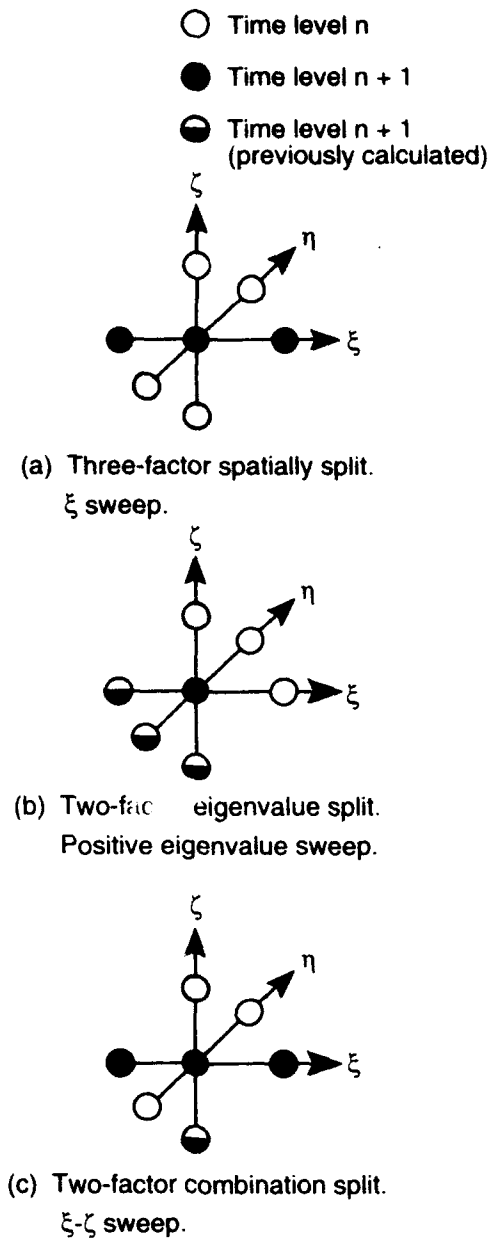


Figure 2.4.3 Computational molecules for advancing the solution in time using approximate factorization schemes.

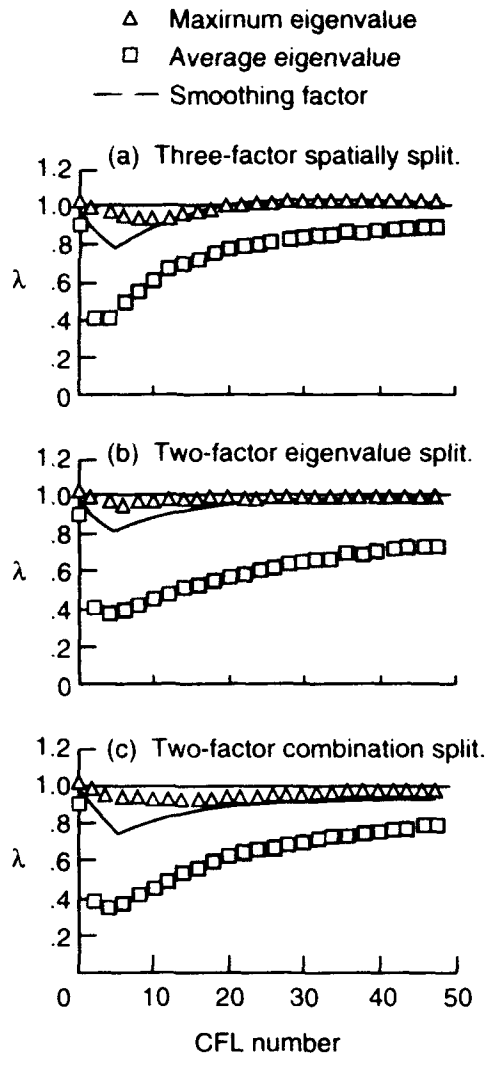


Figure 2.4.4 Stability analysis of three-dimensional approximate factorization schemes; $M_\infty = 0.8; \alpha = 0°$.

Since the implicit spatial discretizations of two of the coordinate directions are not required to be split, either central or upwind differencing can be used in those directions; it is a simple matter to extend the method to include the corresponding thin-layer viscous terms. The computational molecule is shown in Fig. 2.4.3(c). The linearized stability analysis of the coupled Euler equations[40] is shown in Fig. 2.4.4(c), the algorithm retains unconditional stability. The operation count for the implicit inversion is 71 percent[40] of that of the spatially split scheme, as given in Table 2.4.1, and the computations can be vectorized, although the vector lengths are smaller than the spatially split scheme.

Hybrid relaxation-factorization schemes have also been developed to improve the stability characteristics of the three-factor ADI scheme.[56,57] Relaxation is applied along one coordinate direction only (generally streamwise) and approximate factorization is applied in the other two directions. The resulting matrix equation to be solved in each crossflow plane can be written for relaxation in the ξ direction as

$$\begin{aligned}&\{\mathbf{I}+h(|\mathbf{A}|+\delta_\eta\mathbf{B})\}\{\mathbf{I}+h(|\mathbf{A}|)\}^{-1}\\&\{\mathbf{I}+h(|\mathbf{A}|+\delta_\zeta\mathbf{C})\}\,\Delta\mathbf{w}^{l}=\tilde{\mathbf{R}}\left(\mathbf{w}^{l},\mathbf{w}^{l-1}\right)\end{aligned}\tag{2.4.67}$$

where the right-hand-side indicates the nonlinear updating of the residual while sweeping back and forth in the ξ direction. Each factor has the same block tridiagonal or pentadiagonal form of the spatially split scheme and all the operations can be vectorized. The hybrid scheme, termed an AF-SPGS (Symmetric Planar Gauss-Seidel) scheme,[56] avoids the $(\Delta t)^3$ splitting error common to the three-factor schemes and is unconditionally stable for a scalar wave equation. The thin-layer viscous terms can be incorporated easily in the two coordinate directions perpendicular to relaxation since the operators can remain unsplit in those directions.

Minimum-Residual Methods

A class of algorithms that can be very effective and is applicable to both structured– and unstructured–grid methods are minimum-residual methods. For non-symmetric matrices that appear in the solution to the Euler equations, methods such as Conjugate Gradient Squared (CGS),[58] BiConjugate Gradient Stabilized (Bi-CGSTAB),[59] and the Generalized Minimum Residual Method (GMRES)[60] are often used. One of the most widely used and most reliable procedures is GMRES which is briefly described below.

The foundation of GMRES is the projection of the residual error, due to an approximate solution, onto an orthonormal basis for the Krylov subspace which is of smaller dimension than the original problem. Arnoldi's method,[61] which is essentially a Gram-Schmidt procedure, is used to generate an orthonormal basis for the Krylov subspace of dimension k defined as

$$\mathbf{K}_k=\text{span}\left(\bar{\mathbf{r}}_0,\mathbf{A}\bar{\mathbf{r}}_0,\mathbf{A}^2\bar{\mathbf{r}}_0,\ldots,\mathbf{A}^{k-1}\bar{\mathbf{r}}_0\right)\tag{2.4.68}$$

where $\mathbf{r}_0$ is the residual of the initial guess ($\mathbf{r}_0=\mathbf{A}\mathbf{x}_0+\mathbf{b}$) and $\bar{\mathbf{r}}_0=\mathbf{r}_0/\|\mathbf{r}_0\|$. The resulting vectors are orthonormal and satisfy the relation

$$\mathbf{A}\mathbf{V}_k=\mathbf{V}_{k+1}\tilde{\mathbf{H}}\tag{2.4.69}$$

where $\mathbf{V}_{k+1}$ is a $N\times(k+1)$ matrix formed from the $k+1$ vectors obtained after k steps of Arnoldi's method and $\mathbf{H}$ is a $(k+1)\times k$ upper Hessenberg matrix.

For solving a linear system of N equations in N unknowns the final solution is given by

$$\mathbf{x}=\mathbf{x}_0+\mathbf{z}\tag{2.4.70}$$

where $\mathbf{x}_0$ is an initial approximation to the solution and $\mathbf{z}$ is a correction to the initial guess that satisfies

$$\mathbf{A}\mathbf{z}+\mathbf{r}_0=0\tag{2.4.71}$$

Considering $\mathbf{z}$ as an approximate solution to Eq. (2.4.71), then the error is made to be orthogonal to all vectors in $\mathbf{V}_{k+1}$

$$\mathbf{V}_{k+1}^{T}\left(\mathbf{A}\tilde{\mathbf{z}}+\mathbf{r}_0\right)=0\tag{2.4.72}$$

Assuming that $\mathbf{z}$ lies in the k dimensional Krylov subspace such that

$$\mathbf{z}=\mathbf{V}_k\mathbf{y}=y_1\mathbf{v}_1+y_2\mathbf{v}_2+\cdots+y_k\mathbf{v}_k\tag{2.4.73}$$

then utilizing Eqs. (2.4.69) and choosing the initial vector in the Krylov subspace to be the initial residual divided by its magnitude, Eq. (2.4.72) can be written as

$$\mathbf{H}\mathbf{y}+\|\mathbf{r}_0\|\mathbf{e}_1=0\tag{2.4.74}$$

where $\mathbf{e}_1$ is a column vector with unity as its first element and zero for all the remaining elements. Eq. (2.4.74) represents a system of $k+1$ equations with k unknowns which can be solved with a least squares procedure using Givens' rotations. Note that if the error had been projected onto a subspace of vectors of dimension k instead of $k+1$ in Eq. (2.4.72) this would yield a system of k equations in k unknowns which could be solved using standard elimination techniques and is referred to as the full orthogonalization method.[61] However, the least-square approach allows an efficient stopping criteria since the last element in the right-hand-side vector after applying the Givens rotations will represent the error in the solution.

Note that $\mathbf{v}_{k+1}$ is computed during the k^{th} step of the algorithm and will be nonzero unless the exact solution obtained. This leads to an important property of GMRES in that the algorithm can break down only after the exact solution is obtained. Another related property is that when solving an $N\times N$ matrix equation, the exact solution will be obtained in at most N steps.

The disadvantages of the GMRES algorithm are that large memory requirements are required for the Arnoldi procedure because each vector must be stored in order to orthogonalize each new vector with respect to all the previous vectors using a modified Gram-Schmidt process. This also increases the computational time as more vectors are accumulated. To circumvent these problems the procedure is usually "restarted" by using a fixed number of vectors to obtain an approximate solution which is then used as an initial guess from which the process can be repeated.

The GMRES procedure is usually applied to a "preconditioned" system of equations such as

$$\mathbf{M}_1^{-1}\mathbf{A}\mathbf{M}_2^{-1}\mathbf{M}_2\mathbf{x}=\mathbf{M}_1^{-1}\mathbf{b}\tag{2.4.75}$$

where $\mathbf{M}_1$ and $\mathbf{M}_2$ are called the left and right preconditioners, respectively. The role of the preconditioners is to obtain a more favorable distribution of eigenvalues than the original system in order to obtain faster convergence so that suitable accuracy can be obtained while reducing the dimension of the Krylov subspace. In practice, the success of using GMRES depends very strongly on the effectiveness of the preconditioners.

The GMRES procedure, as originally developed and as described above, is for solving a linear system of equations. Wigton et al.[62] have extended this algorithm for accelerating the convergence of nonlinear equations such as the Euler equations with good success. This procedure is described in section 2.5.5.

Extensions to Unstructured Grids

With unstructured grids, no regular a priori ordering of the grids exists such that the matrix equation can be factored explicitly as a product of simpler matrices, as in the ADI algorithms. Hence, the general approach is to solve the matrix equation with a direct method or iteratively with relaxation methods.[63-65] Consider as the starting point the linearized single-step backward-Euler time differencing scheme, Eq. (2.4.45), written as

$$[\mathbf{A}]^n\{\Delta\mathbf{w}\}^n = \{\mathbf{R}\}^n \tag{2.4.76}$$

where

$$[\mathbf{A}]^n = \frac{\mathbf{I}}{J\Delta t} + \frac{\partial \mathbf{R}^n}{\partial \mathbf{w}} \tag{2.4.77}$$

The solution of Eq. (2.4.76) can, in principle, be obtained by a direct inversion of $[\mathbf{A}]^n$. This technique, while quite successful in two dimensions, is currently not very feasible for practical calculations in three dimensions.

Instead, first-order-accurate upwind approximations on the left hand side of Eq. (2.4.76) can be utilized in order to reduce both the bandwidth of the equations and thereby the required storage, as discussed previously. The resulting scheme can retain the desirable property of stability for large time steps, for either first or second order spatial differencing of the residual, if the linear system is solved to a sufficiently low tolerance.

Now consider a sample configuration of triangles in which the cells are randomly ordered, shown in Fig. 2.4.5. The corresponding form of the matrix $[\mathbf{A}]^n$ is shown in Fig. 2.4.6 where a circle represents the nonzero entries.

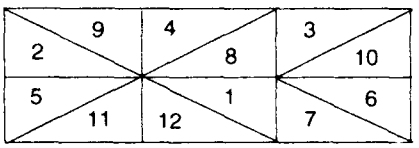


Figure 2.4.5 Sample Cell Configuration.

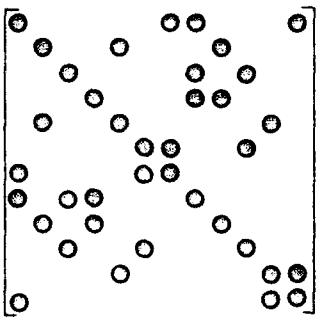


Figure 2.4.6 Form of Matrix for Cells in Fig. 2.4.5

A variety of relaxation schemes can be constructed in which the solution of Eq. (2.4.76) is obtained through a sequence of iterations in which an approximation of $\Delta\mathbf{w}$ is continually refined.

To facilitate the derivation of these schemes, $[\mathbf{A}]^n$ is first written as a linear combination of three matrices representing the diagonal, subdiagonal, and superdiagonal terms, i.e.,

$$[\mathbf{A}]^n = [\mathbf{D}]^n + [\mathbf{M}]^n + [\mathbf{N}]^n \tag{2.4.78}$$

The simplest iterative scheme for obtaining a solution to the linear system of equations is a Jacobi type method in which all the off-diagonal terms of $[\mathbf{A}]^n\{\Delta\mathbf{w}\}$ (i.e., $[\mathbf{M}]^n\{\Delta\mathbf{w}\} + [\mathbf{N}]^n\{\Delta\mathbf{w}\}$), are taken to the right-hand side of Eq. (2.4.76) and are evaluated using the values of $\{\Delta\mathbf{w}\}^i$ from the previous subiteration level i. This scheme can be represented as

$$\begin{aligned}[\mathbf{D}]^n\{\Delta\mathbf{w}\}^{i+1} &= \left[\{\mathbf{R}\}^n - [\mathbf{M}+\mathbf{N}]^n\{\Delta\mathbf{w}\}^i\right] \\ &= \left[\{\mathbf{R}\}^n - [\mathbf{O}]^n\{\Delta\mathbf{w}\}^i\right]\end{aligned} \tag{2.4.79}$$

where $[\mathbf{O}]^n$ denotes the off-diagonal terms in the matrix. The disadvantage of the above scheme is that the sequence of Jacobi iterations may converge somewhat slowly. In order to accelerate the convergence, a Gauss-Seidel procedure may be employed in which values of $\{\Delta\mathbf{w}\}$ are used on the right-hand side of Eq. (2.4.79) as soon as they are available. An example of this scheme, corresponding to a sequential solution of the equations from the first to the last element, can be written as

$$[\mathbf{D}]\{\Delta\mathbf{w}\}^{i+1} = \left[\{\mathbf{R}\}^n - [\mathbf{M}]^n\{\Delta\mathbf{w}\}^{i+1} - [\mathbf{N}]^n\{\Delta\mathbf{w}\}^i\right] \tag{2.4.80}$$

where the latest values of $\{\Delta\mathbf{w}\}$ associated with the subdiagonal terms are immediately used on the right hand side of the iteration equation. A slight modification to the above algorithm in which the latest values of $\{\Delta\mathbf{w}\}$ associated with the superdiagonal are used, corresponding to a sequential solution from the last to the first element, results in a very similar scheme which is given by

$$[\mathbf{D}]\{\Delta\mathbf{w}\}^{i+1} = \left[\{\mathbf{R}\}^n - [\mathbf{M}]^n\{\Delta\mathbf{w}\}^i - [\mathbf{N}]^n\{\Delta\mathbf{w}\}^{i+1}\right] \tag{2.4.81}$$

A symmetric Gauss-Seidel type procedure is obtained by alternating the use of Eq. (2.4.80) with Eq. (2.4.81).

Note that the algorithms given above by Eqs. (2.4.80) and (2.4.81) can both be implemented by sweeping sequentially

through each mesh cell and simply using the latest values of $\{\Delta \mathbf{w}\}$ for all the off-diagonal terms which have been taken to the right hand side. This can be represented as

$$[\mathbf{D}]\{\Delta \mathbf{w}\}^{i+1} = \left[\{\mathbf{R}\}^{n} - [\mathbf{O}]^{n}\{\Delta \mathbf{w}\}^{i+\frac{1}{2}}\right] \qquad (2.4.82)$$

where $\mathbf{w}^{i+\frac{1}{2}}$ is the most recent value of $\mathbf{w}$ and will be at subiteration level $i+1$ for the cells which have been previously updated and at level i for the cells which remain to be updated. The distinction between algorithms (2.4.80) and (2.4.81) comes about from the sequential solution of the equations in opposite directions (forward and backward, respectively) through the elements.

There are two disadvantages of the scheme as described above. The first disadvantage is that the process is not vectorizable, since the solution of each element must be obtained before proceeding to the next one. The second disadvantage of this scheme is that the degree of implicitness is set by the ordering of the elements. This can be illustrated by noting that although the off-diagonal terms may be updated and immediately used on the right-hand side, the solution of the next unknown may or may not depend on previously determined quantities. For example, as can be seen from Fig. 2.4.5, when solving for unknown number 2 using Eq. (2.4.80), the updated value of the solution at element 1 is not used so the solution for point 2 remains a Jacobi step.

Note that for structured grids in which the cells are ordered in a natural manner (e.g., left to right and top to bottom), the latest information will be used immediately for calculation of the next unknown. This is because the ordering of the cells produces a banded matrix with terms grouped along the diagonal. The fact that the latest available data is not necessarily used for updating information in unstructured grids is strictly due to the random ordering of the cells.

An improvement can be obtained by simply renumbering the cells in such a way as to group terms along the diagonal of the matrix. In this manner, the solution of each point will tend to ensure that previously updated information from the surrounding cells is used as soon as it is available. An example of this is shown in Fig. 2.4.1 where the same sample set of cells used in Fig. 2.4.5 is simply renumbered from bottom to top and left to right. The resulting form of the matrix, shown in Fig. 2.4.8, shows that the grouping along the diagonal is greatly improved. The ordering of the cells in this way results in a faster convergence of the linear system than a random ordering of cells. For general unstructured-grid coefficient matrices, the bandwidth reduction algorithms discussed previously are effective in clustering unknowns along the diagonal. Again, it should be noted that several variations of this scheme can be obtained by using various combinations of Eqs. (2.4.80) and (2.4.81). An important disadvantage of this scheme, however, is that it still suffers from the fact that the contribution of the off-diagonal terms to the right-hand side of Eq. (2.4.82) is not vectorizable.

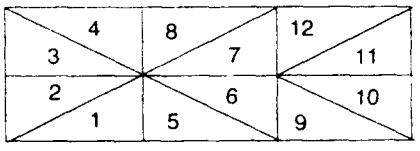


Figure 2.4.7 Sample Cells.

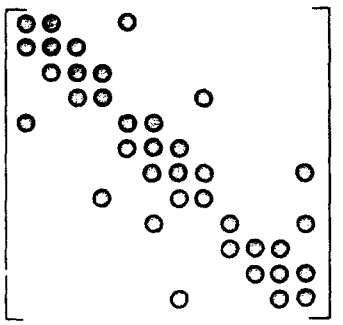


Figure 2.4.8 Form of Matrix for Cells in Fig. 2.4.1

The Jacobi, Gauss-Seidel, and symmetric Gauss-Seidel schemes described above have all been used in practice by various researchers. Applications[64] to a circular arc in a channel indicated that the symmetric Gauss-Seidel scheme exhibited the fastest convergence rate of these three schemes. The symmetric Gauss-Seidel algorithm of Batina,[65] applied to transonic flow over airfoils, enhanced the grouping of the unknowns along the diagonal by sorting them according to the x coordinate direction. Variants of line-relaxation schemes can be constructed by ordering the elements into groups of approximately collinear elements and then updating the elements by groups. Note that in the particular case above corresponding to the use of first-order upwind implicit differencing, the relaxation is applied to solve the linear system of equations and, as a consequence, no particular sweeping directions need be maintained, such as symmetric Gauss-Seidel. However, if the relaxation is applied directly to the nonlinear implicit equation, where the residual equation is evaluated at every iteration, then the stability of the scheme is coupled to the discretization of both the implicit operator and the spatial discretization. In that case, a Von Neumann stability analysis indicates that a second-order spatial differencing of the residual requires an alternating Gauss-Seidel scheme in order to remain stable.

A numbering of the cells which has proved useful on vector processing computers is shown in Fig. 2.4.9. The ordering is obtained by grouping cells so that no two cells in a given group share a common edge. The resulting matrix form for $[\mathbf{A}]$ is given in Fig. 2.4.10. Note that for the current example, only two groups are formed: the first group consists of the cells numbered 1-6, and the second group contains cells 7-12. In practice, four groups are generally sufficient for two dimensional calculations and five groups for three dimensional calculations.

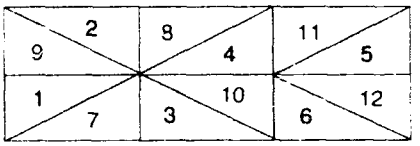


Figure 2.4.9 Sample Cells.

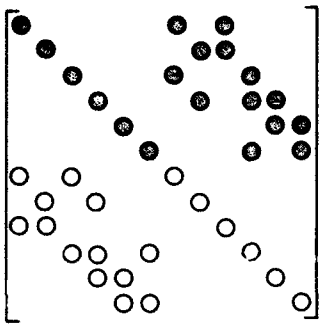


Figure 2.4.10 Form of Matrix for Cells in Fig. 2.4.9.

The solution scheme can be written as before using Eq. (2.4.82) and is implemented by solving for all the unknowns in a group at a time. In this manner, the cells in the first group are solved using a Jacobi type iteration while the cells in all the subsequent groups are obtained by using the most recently updated values of $\{\Delta \mathbf{w}\}$ from the off-diagonal contributions. In this way, a Gauss-Seidel type scheme is obtained which is easily implemented and is fully vectorizable. The relaxation can be considered as a generalization of the checkerboard relaxation schemes used for structured grids with a three-point stencil in each coordinate directions. The points are colored in a checkerboard fashion as either red or black; all the red points are updated simultaneously, followed by all the black points, and so forth. Note that in the discussions above, the exact number of subiterations required in order to sufficiently converge the linear problem (Eq. (2.4.76)) has not been specified. The optimum number of subiterations used for each global time step is generally determined through numerical experiments and is problem dependent.[63]

2.4.3 References

1. HIRSCH, C., *Numerical Computation of Internal and External Flows, Volume I*, Wiley and Sons, Chichester, UK, 1989.

2. LAX, P. D., "Weak Solutions of Nonlinear Hyperbolic Equations and their Numerical Computation," Comm. Pure and Applied Mathematics, 7,159-193,1954.

3. LAX, P. D. and WENDROFF B., "Systems of Conservation Laws," Comm. Pure and Applied Mathematics, 13, 217-237, 1960.

4. MacCORMACK, R. W., "The Effect of Viscosity in Hypersonic Impact Cratering," AIAA 69-354, 1969.

5. MacCORMACK, R. W.,"A Numerical Method for Solving the Equations of Compressible Viscous Flow," AIAA Journal, Vol. 20, No. 9, 1275-1281, 1969.

6. SCHWARZ, H. R., *Numerical Analysis*. Wiley and Sons, Chichester, UK, 1988.

7. JAMESON, A., SCHMIDT W. and TURKEL E., "Numerical Solutions of the Euler Equations by Finite Volume Methods using Runge-Kutta Time Stepping Schemes," AIAA 81-1259, 1981.

8. JAMESON, A., "Successes and Challenges in Computational Aerodynamics," AIAA 87-1184-CP, 1987.

9. RIZZI, A. W., "Damped Euler Equation Method to Compute Transonic Flow around Wing-Body Combinations," AIAA Journal, Vol. 20, No. 10, Oct. 1982.

10. JAMESON, A., "Using Euler Schemes," AIAA Professional Study Series, Snowmass, CO, June 1985.

11. VENKATAKRISHNAN, V., "Viscous Computations Using a Direct Solver," Computers and Fluids, Vol. 18, no. 2, pp. 191-204, 1990.

12. BAILEY, H. E. and BEAM, R. M., "Newton's Method Applied to Finite-Difference Approximations for the Steady-State Compressible Navier-Stokes Equations," Journal of Computational Physics, Vol. 93, 1991, pp. 108-127.

13. BEAM, R. M. and WARMING, R. F., "An Implicit Factored Scheme for the Compressible Navier-Stokes Equations," AIAA Journal, Vol. 16, No. 4, April 1978, pp. 393-402.

14. JESPERSON, D. C. and PULLIAM, T. H., "Flux Vector Splitting and Approximate Newton Methods," AIAA 83-1899-CP, July, 1983.

15. ROBERTS, T. and WARREN, G., "Analysis of Implicit Second-Order Upwind-Biased Stencils," AIAA 93-3379, July 1993.

16. BRILEY, W. R. and MCDONALD, H., "On the Structure and Use of Linearized Block Implicit Schemes," Journal of Computational Physics, Vol. 34, pp. 54-73, 1980.

17. STEGER, J. L. and WARMING, R. F., "Flux Vector Splitting of the Inviscid Gasdynamic Equations with Applications to Finite-Difference Methods," Journal of Computational Physics, Vol. 40, No. 2, April 1981, pp. 263-293.

18. DUFF, I. S., ERISMAN, A. M. and REID, J. K., *Direct Methods for Sparse Matrices. Monographs on Numerical Analysis*. Clarendon Press, Oxford, 1986.

19. EISENSTADT, S. C., BURSKY, M.C., SCHULTZ, M. H. and SHERMAN, A. H., "Yale Sparse Matrix Package: II The Nonsymmetric Codes," Research Report No. 114, Dept. of Computer Sciences, Yale University, 1977.

20. GEORGE, A. and LIU, J. W., *Computer Solutions of Large Sparse Positive Definite Systems*. Prentice-Hall, 1981.

21. VENKATAKRISHNAN, V. and BARTH, T., "Application of Direct Solvers to Unstructured Meshes for the Euler and Navier-Stokes Equations Using Upwind Schemes," AIAA 89-0364, 1989.

22. RIGGINS, D. W. and WALTERS, R. W., "Direct Solutions for the Computation of Compressible Flows," AIAA 88-0229, 1988.

23. JAMESON, A. and TURKEL, E., "Implicit Schemes and LU Decompositions," Mathematics of Computation, Vol. 37, pp. 385-397, 1981.

24. RIEGER, H. and JAMESON, A., "Solution of Steady Three-Dimensional Compressible Euler and Navier-Stokes Equations by an Implicit LU Scheme," AIAA 88-0619, Jan. 1988.

25. PULLIAM, T. H. and CHAUSSEE, D. S., "A Diagonal Form of an Implicit Approximate Factorization Algorithm," Journal of Computational Physics, Vol. 39, pp. 347-363, 1981.

26. OBAYASHI, S., MATSUSHIMA, K., FUJII, K. and KUWAHARA, K., "Improvements in Efficiency and Reliability for Navier-Stokes Computations Using the LU-ADI Factorization Algorithm," AIAA 86-338, Jan. 1986.

27. LOMBARD, C. K., BARDINA, J., VENKATAPATHY, E. and OLIGER, J., "Multi-Dimensional Formulation of CSCM - an Upwind Flux Difference Eigenvector Split Method for the Compressible Navier-Stokes Equations," AIAA 83-1895-CP, July 1983.

28. NAGASU, H. and FUJII, K., "Navier-Stokes Code Development for Transonic Flow Simulations," *Computational Methods in Viscous Aerodynamics*, Ed. T.K.S. Murthy and C.A. Brebbia, Elsevier and Computational Mechanics Publications, 1990, pp. 273-300.

29. PULLIAM, T. H., "Euler and Thin Layer Navier Stokes Codes: ARC2D and ARC3D," Notes for Computational Fluids Dynamics Workshop, University of Tennessee Space Institute, March 1984.

30. ANDERSON, W. K., THOMAS, J. L. and WHITFIELD, D. L., "Three-Dimensional Multigrid Algorithms for the Flux-Split Euler Equations," NASA TP-2829, Nov. 1988.

31. SOUTH, J. C., Jr., "Recent Advances in Computational Transonic Aerodynamics," AIAA 85-0366, 1985.

32. WHITFIELD, D., "Newton-Relaxation Schemes for Nonlinear Hyperbolic Equations," MSSU-EIRS-ASE-90-3, Mississippi State University, 1990.

33. GATLIN, B. and WHITFIELD, D., "An Implicit, Upwind, Finite-Volume Method for Solving the Three-Dimensional Thin-Layer Navier-Stokes Equations," AIAA 87-1149CP, June, 1987.

34. CHAKRAVARTHY, S. R., "Relaxation Methods for Unfactored Implicit Upwind Schemes," AIAA 84-0165, Jan. 1984.

35. VAN LEER, B. and MULDER, W. A., "Relaxation Methods for Unfactored Implicit Schemes," Report 84-20, Delft University of Technology, 1984.

36. CHAKRAVARTHY, S. R. and SZEMA, K-Y., "An Euler Solver for Three-Dimensional Supersonic Flows with Subsonic Pockets," Journal of Aircraft, Vol. 24, No. 2, Feb. 1987, pp. 73-83.

37. WALTERS, R. W. and DWOYER, D. L., "An Efficient Iteration Strategy for the Solution of the Euler Equations," AIAA 85-1529, July 1985.

38. THOMAS, J. L., VAN LEER, B. and WALTERS, R. W., "Implicit Flux-Split Schemes for the Euler Equations," AIAA Journal, Vol. 28, No. 6, pp. 973-974, 1990.

39. MacCORMACK, R. W., "Current Status of Numerical Solutions of the Navier-Stokes Equations," AIAA 85-0032, 1985.

40. YOON, S. and KWAK, D., "Implicit Methods for the Navier-Stokes Equations," Computing Systems in Engineering, Vol. 1, Nos. 2-4, pp. 535-547, 1990.

41. CHAKRAVARTHY, S. R., SZEMA, K-Y., GOLDBERG, U. C., GORSKI, J. J. and OSHER, S., "Application of a New Class of High Accuracy TVD Schemes to the Navier-Stokes Equations," AIAA 85-0165, Jan. 1985.

42. EBERLE, A., SCHAFER, O. and SCHMATZ, M. A., "High Order Solutions of the Euler Equations Using Characteristic Flux Averaging," 12th Congress of the International Council of the Aeronautical Sciences, London, England, ICAS-86-1.3.1, 1986.

43. EBERLE, A. "Characteristic Flux Averaging Approach to the Solution of Euler's Equations," VKI Lecture Series 1987-04, Von Karman Institute, Brussels, Belgium, March 1987.

44. BRENNEIS, A. and EBERLE, A., "Unsteady Transonic Flows Past Airfoils Using a Fast Implicit Godunov Type Euler Solver," Proceedings of the 7th GAMM Conference on Numerical Methods in Fluid Mechanics, Ed. Deville M., Notes on Numerical Fluid Mechanics, Volume 20, Friedr. Vieweg & Sohn, Braunschweig/Wiesbaden, 1987.

45. BRENNEIS, A., EBERLE, A. and SCHMATZ, M. A., "Verification of an Implicit Relaxation Method for Steady and Unsteady Viscous and Inviscid Flow Problems," AGARD Symposium on Validation of Computational Fluid Dynamics, Lisbon, Portugal, 1988.

46. BRENNEIS, A. and EBERLE, A., "Unsteady Transonic Flows Past Airfoils and Wings Using a Fast Implicit Godunov Type Euler Solver," ICAS-paper 88.6.3.1, 16th Congress of the International Council of the Aeronautical Sciences, Jerusalem, Israel, 1988.

47. BRENNEIS, A. and EBERLE, A., "Application of an Implicit Relaxation Method Solving the Euler Equations for Real Unsteady Problems," International Symposium on Unsteady Fluid Dynamics, Toronto, Canada, 1990.

48. BRENNEIS, A. and EBERLE, A., "Application of an Implicit Relaxation Method Solving the Euler Equations for Time-Accurate Unsteady Problems," Journal of Fluids Engineering, Vol. 112, No. 4, Dec. 1990, pp. 510-520.

49. BRENNEIS, A. and EBERLE, A., "Evaluation of an Unsteady Implicit EULER Code Against Two and Three-Dimensional Standard Configurations," AGARD Specialists' Meeting on Transonic Unsteady Aerodynamics and Aeroelasticity, Paper No. 10, AGARD CP 507, 1991.

50. EBERLE, A., RIZZI, A. and HIRSCHEL, E. H., *Numerical Solutions of the Euler Equations for Steady Flow Problems*, Notes on Numerical Fluid Mechanics, Vieweg, Braunschweig/Wiesbaden, Vol. 34, 1992.

51. EBERLE, A. and HEISS, S., "Enhanced Numerical Inviscid and Viscous Fluxes for Cell Centered Finite Volume Schemes, Computers and Fluids, Pergamon Press, Vol. 22, No. 2/3, 1993, pp. 295-309.

52. KRAEMER, E., HERTEL, J. and WAGNER, S., "Computation of Subsonic and Transonic Helicopter Rotor Flow Using Euler Equations," 13th European Rotorcraft Forum, Arles, France, Paper 2–14, 1987.

53. KRAEMER, E., HERTEL, J. and WAGNER, S., "Euler Procedure for Calculation of the Steady Rotor Flow with Emphasis on Wake Evolution," AIAA 90–3007, 1990.

54. BARTH, T. J., "Analysis of Implicit Local Linearization Techniques for Upwind and TVD Algorithms," AIAA 87-0595, January, 1987.

55. YING, S. X., STEGER, J. L., SCHIFF, L. B. and BAGANOFF, D.,"Numerical Simulation of Unsteady, Viscous, High-Angle-of-Attack Flows Using a Partially Flux-Split Algorithm," AIAA 86-2179, Aug. 1986.

56. HARTWICH, P-M, HSU, C-H. and LIU, C. H.,"Vectorizable Implicit Algorithms for the Flux-Difference Split, Three-Dimensional Navier-Stokes Equations," Journal of Fluids Engineering, Vol. 110, Sept. 1988, pp. 297-305.

57. THOMAS, J. L. and WALTERS, R. W., "Upwind Relaxation Algorithms for the Navier-Stokes Equations," AIAA Journal, Vol. 25, No. 4, April 1987, pp. 527-531.

58. SONNEVELD, P. "CGS, A Fast Lanczos-Type Solver for Nonsymmetric Linear Systems," Siam J. Sci. Stat. Comput. Vol 10, No. 1, January 1989.

59. VAN DER VORST, H. A., "BI-CGSTAB: A Fast and Smoothly Converging Variant of BI-CG for the Solution of Nonsymmetric Linear Systems," SIAM J. Sci. Stat. Comput. Vol 13, No. 2 March 1992.

60. SAAD, Y. and SCHULTZ, M. H., "GMRES: A Generalized Minimal Residual Algorithm for Solving Nonsymmetric Linear Systems," SIAM J. Sci. Stat. Comput. Vol. 7, No 3, July, 1986.

61. ARNOLDI, W. E., "The Principle of Minimized Iteration in the Solution of the Matrix Eigenvalue Problem," Quart. Appl. Math., 9, 1951.

62. WIGTON, L., YU, N. and YOUNG, D. "GMRES Acceleration of Computational Fluid Dynamics Codes," AIAA 85–1494, 1985.

63. ANDERSON, W. K., "Grid Generation and Flow Solution Method for Euler Equations on Unstructured Grids," NASA Technical Memorandum 4295, April 1992.

64. WHITAKER, D. L., SLACK, D. C. and WALTERS, R. W., "Solution Algorithms for the Two-Dimensional Euler Equations on Unstructured Meshes," AIAA 90–0067, 1990.

65. BATINA, J., "Implicit Flux-Split Euler Schemes for Unsteady Aerodynamic Analyses Involving Unstructured Dynamic Meshes," AIAA 90–0936, 1990.

2.5 CONVERGENCE ACCELERATION

The acceleration of the convergence rate of numerical computations is very important when conducting grid convergence studies to verify the accuracy of numerical solutions. An integral part of this process is the uniform refinement of the grid in each direction until little or no variation in the solution is observed with increased grid size. Unfortunately, without convergence acceleration, the convergence rate of iterative methods severely degrades as the grid spacing is decreased through the grid refinement process. To mitigate the penalties associated with the use of fine grids, several methods of convergence acceleration have been introduced and are discussed below. These methods are especially important in three dimensions, where an eight-fold increase in the number of grid points occurs when the points are doubled in all three coordinate directions.

2.5.1 Local Time Stepping

One of the simplest and most commonly used methods of convergence acceleration is the use of local time stepping.[1] When a steady-state solution is of interest, a spatially varying step size can be used for each cell independent of the other cells. The time step is generally based on a combination of the flow variables in each cell as well as the cell size.

Perhaps the most commonly used method of local time stepping is to base the time step in each cell on a local CFL number. Examples of this can be found in many references, as can be seen in the citations of the code summaries in chapter 3, in which the time step in each cell is determined by

$$\Delta t_{cell} = \mathrm{CFL} \cdot \Delta t_{(\mathrm{CFL}=1)} \tag{2.5.1}$$

where $\Delta t_{(\mathrm{CFL}=1)}$ is the time step required for a CFL of unity and may be determined with a variety of definitions for multidimensional problems. One form that is commonly used is given by

$$\Delta t_{(\mathrm{CFL}=1)} = \frac{2V}{\int (|\mathbf{v}\cdot\hat{\mathbf{n}}| + a)\,dA} \tag{2.5.2}$$

where the integral is evaluated over the surface of each control volume.

An example of the effectiveness of local time stepping is shown in Fig. 2.5.1, where an explicit method is used to compute the flow around an NACA 0012 airfoil at a Mach number of 0.8 and an angle of attack of 1.25°. The effect of local time stepping is dramatic. By using local time stepping, the residual is reduced 10 orders of magnitude in approximately 3000 iterations; without local time stepping, little progress towards convergence is achieved. The effect is even more dramatic on the lift coefficient, where, with local time stepping, the final lift value is obtained in about 500 iterations. The solution obtained without local time stepping has failed to reach a steady state after 3000 iterations. This technique is very simple and easily implemented and offers a clear advantage toward acceleration of the solution to a steady state.

2.5.2 Residual Smoothing

To accelerate the convergence of explicit algorithms, one methodology that has been extremely effective is residual smoothing.[2,3] For this method, the steady-state residual calculated at each step is modified in such a way that the support of the [illegible] is enhanced, which increases the implicitness of the [illegible]. In practice, this technique has been particularly [illegible] for central-differencing schemes when used in conjunction with multistage time stepping, although recent improvements for upwind discretizations have been reported.[4] For this reason, the general procedure is outlined below for a four-stage Runge-Kutta type of algorithm, applied to a one-dimensional model problem with central differencing. The effect of residual smoothing on the stability is examined through the application of a Fourier analysis.

Consider the model problem given by

$$u_t + u_x + \mu \Delta x^3 u_{xxxx} = 0 \tag{2.5.3}$$

A four-stage Runge-Kutta type method is given by

$$\begin{aligned} u^{(0)} &= u^{(n)} \\ u^{(1)} &= u^{(0)} - \alpha_1 \Delta t R^{(0)} \\ u^{(2)} &= u^{(0)} - \alpha_2 \Delta t R^{(1)} \\ u^{(3)} &= u^{(0)} - \alpha_3 \Delta t R^{(2)} \\ u^{(4)} &= u^{(0)} - \Delta t R^{(3)} \\ u^{(n+1)} &= u^{(4)} \end{aligned} \tag{2.5.4}$$

where $R^{(i)} = \delta_x u^{(i)} + \mu \Delta x^3 \delta_{xxxx} u^{(i)}$ denotes the discretized steady-state residual formed from data at stage level i. Note

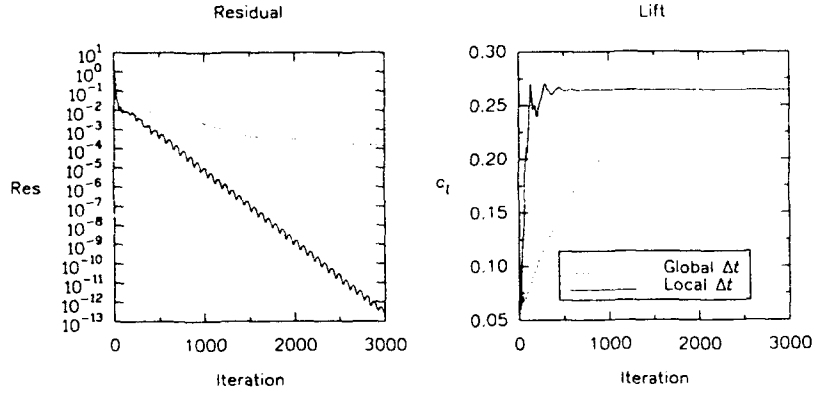


Figure 2.5.1 Effect of local time stepping on convergence rate.

that in the present form this scheme is one in which the dissipative term is evaluated at each stage. It is possible (and more economical) to use schemes in which the dissipation is only evaluated periodically, for example on the first and third stages.[1] However, for illustrative purposes, the dissipation will be evaluated on each stage.

To determine the stability of the current scheme, a Fourier mode is substituted for u:

$$u = \hat{U} e^{i\xi x} \tag{2.5.5}$$

The Fourier symbol for $\Delta t R^{(1)}$ is now written as

$$\Delta t \hat{R}^{(1)} = -\hat{U}^{(1)} e^{i\xi x} Z \tag{2.5.6}$$

where

$$Z = -\lambda(i \sin\xi + 4\mu(1 - \cos\xi)^2) \tag{2.5.7}$$

and $\lambda = \Delta t/\Delta x$ is the Courant number. Substitution of Eqs. (2.5.5), (2.5.6), and (2.5.7) into Eq. (2.5.4) yields an equation for the amplification factor

$$g = 1 + Z + \alpha_1 Z^2 + \alpha_2\alpha_3 Z^3 + \alpha_1\alpha_2\alpha_3 Z^4 \tag{2.5.8}$$

which indicates the extent to which errors decrease (or grow) from one iteration to the next. Stability of the scheme requires that $|g| \leq 1$ for all Z.

By cycling through values of $0 \leq \xi \leq \pi$, with Eq. (2.5.7) used in Eq. (2.5.8), the amplification factor can be obtained for a fixed value of λ and a dissipation coefficient μ. Figure 2.5.2 shows the amplification factor as a function of ξ with a "standard" set of coefficients given by $\alpha_1 = 1/4$, $\alpha_2 = 1/3$, $\alpha_3 = 1/2$, and $\mu = 1/32$. As shown in the figure, the

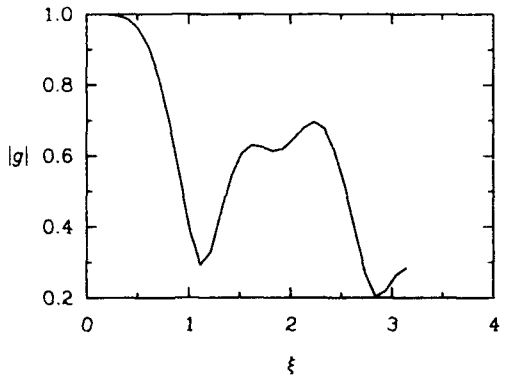


Figure 2.5.2 Amplification factor for standard four–stage scheme; $\lambda = 2.8$, $\alpha_1 = 1/4$, $\alpha_2 = 1/3$, $\alpha_3 = 1/2$, and $\mu = 1/32$.

algorithm with these parameters is indeed stable. If, however, the value of λ is increased to 3, then the scheme becomes unstable near $\xi = \pi/2$, as shown in Fig. 2.5.3.

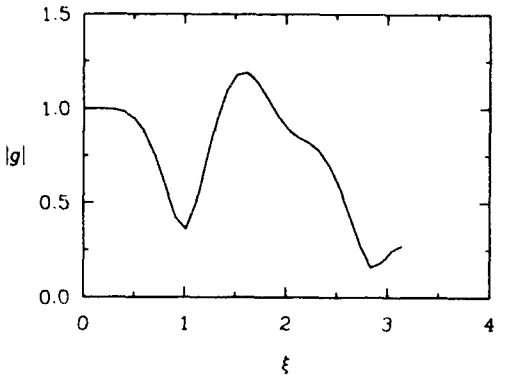


Figure 2.5.3 Amplification factor for standard four–stage scheme; $\lambda = 3.0$, $\alpha_1 = 1/4$, $\alpha_2 = 1/3$, $\alpha_3 = 1/2$, and $\mu = 1/32$.

To obtain a stable algorithm for higher values of λ, the support of the scheme may be increased by replacing the residuals $(\cdots, R_{i-1}, R_i, R_{i+1}, \cdots)$ at each point with an average of the residuals on either side:

$$\overline{R_i} = \epsilon R_{i-1} + (1 - 2\epsilon)R_i + \epsilon R_{i+1} = (1 + \epsilon\delta_{xx})R_i \tag{2.5.9}$$

With this modification to the residual, the Fourier symbol Z is now given by

$$Z = -\lambda[i \sin\xi + 4\mu(1 - \cos\xi)^2][1 - 2\epsilon(1 - \cos\xi)] \tag{2.5.10}$$

With the addition of the second factor, the value of λ may be increased. An example is shown in Fig. 2.5.4 for the four–stage scheme described above, but $\lambda = 3$ is used where the residual is replaced by the averaging procedure in Eq. (2.5.9) with $\epsilon = 1/4$. As shown in the figure, the scheme is now stabilized for $\lambda = 3$, whereas it was previously unstable. Experimentation has shown that for $\epsilon = 1/4$ an increase of approximately 50 percent in the value of λ can be obtained.

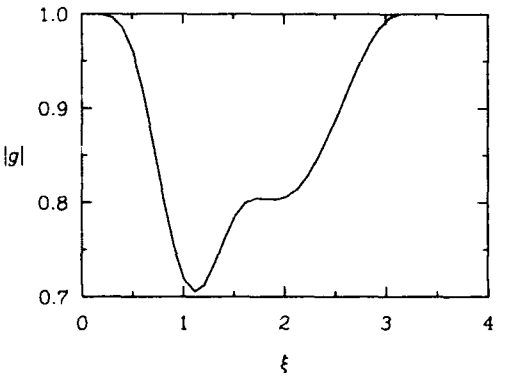


Figure 2.5.4 Amplification factor for standard four-stage scheme with explicit residual smoothing; $\lambda = 3.0$, $\epsilon = 1/4$, $\alpha_1 = 1/4$, $\alpha_2 = 1/3$, $\alpha_3 = 1/2$, and $\mu = 1/32$.

A disadvantage of the above procedure can be illustrated for $\epsilon = 1/4$. For this value of ϵ, if the residuals exhibit an

odd-even type of behavior, the value of $\overline{R}_i$ computed with Eq. (2.5.9) will be zero, so that no update of the dependent variables at each grid point would occur regardless of the value of R_i at that point. To overcome this difficulty and to allow arbitrary values of ϵ, an average residual may be calculated from an implicit relation given by

$$-\epsilon \overline{R}_{i-1} + (1 + 2\epsilon)\overline{R}_i + \epsilon \overline{R}_{i+1} = (1 - \epsilon\delta_{xx})\overline{R}_i = R_i \tag{2.5.11}$$

In this manner, the support of the scheme can be made to extend over the entire grid thus relaxing the time step limitation. The Fourier symbol of the resulting scheme is given by

$$Z = \frac{-\lambda[i \sin\xi + 4\mu(1 - \cos\xi)^2]}{[1 + 2\epsilon(1 - \cos\xi)]} \tag{2.5.12}$$

The denominator is greater than 1 for all values of $\xi > 0$ and reduces the magnitude of Z so that larger time steps can be taken. Reference 5 shows that in the absence of dissipation, stability is maintained for any value of λ if ϵ is chosen so that

$$\epsilon \geq \frac{1}{4}\left[\left(\frac{\lambda}{\lambda^*}\right)^2 - 1\right] \tag{2.5.13}$$

where λ^* is the limit for the original scheme without residual averaging.

The success of this technique is demonstrated in Fig. 2.5.5, where the value of λ^* is assumed to be 2.8, which is the stability limit of the standard four–stage scheme without added dissipation ($\mu = 0$). For example, to achieve stability for $\lambda = 7$, Eq. (2.5.13) indicates that a value of ϵ of $1\frac{1}{2}$ is appropriate. As shown in the figure, the amplification factor remains below unity for all values of $0 \leq \xi \leq \pi$.

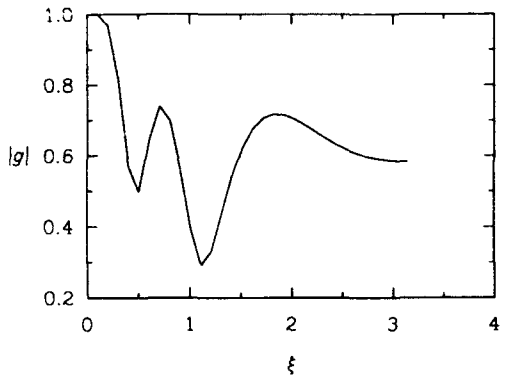


Figure 2.5.5 Amplification factor for standard four-stage scheme with implicit residual smoothing; $\lambda = 7$, $\epsilon = 1\frac{1}{2}$, $\alpha_1 = 1/4$, $\alpha_2 = 1/3$, $\alpha_3 = 1/2$, and $\mu = 1/32$.

To implement Eq. (2.5.11) for multidimensional problems, two prominent methodologies can be used. For structured grids in two dimensions, an approximate factorization procedure is often used, in which the implicit operator is spatially split into a product of two more easily invertible ones.[5]

$$(1 - \epsilon\delta_{xx} - \epsilon\delta_{yy})\overline{R}_i \approx (1 - \epsilon\delta_{xx})(1 - \epsilon\delta_{yy})\overline{R}_i = R_i \tag{2.5.14}$$

This equation is solved in two steps; both steps involve the solution of a tridiagonal system of equations.

Another technique for implementing residual smoothing in multidimensions is to solve the system of equations with a point Jacobi or Gauss-Seidel type of procedure. This technique has been predominantly used for unstructured grid algorithms in which a spatial factorization is not easily achieved;[6] however, the implementation of residual smoothing into structured grid codes has also been achieved in this manner and has yielded advantageous convergence properties over the factored form.[7]

As previously mentioned, the technique of implicit residual smoothing has been extended to include upwind discretizations for the implicit smoothing operator.[4,8] For the one-dimensional model problem, the previously central-differenced smoothing operator given by Eq. (2.5.11) is replaced by an upwind operator

$$-\epsilon \overline{R}_{i-1} + (1 + \epsilon)\overline{R}_i = R_i \tag{2.5.15}$$

With this modification, a significant increase in the allowable CFL number is achieved over the central-difference formulation without destroying desirable smoothing properties of the high-frequency error components. While the implementation of this technique is not straightforward for multidimensional problems because of omnidirectional wave propagation, results in Refs. 4, 7, and 8 for two-dimensional Euler computations indicate significant improvements over residual smoothing with pure central differencing.

2.5.3 Vector Sequence Extrapolation

Sequence Acceleration

Vector-sequence extrapolation is a well-known technique for accelerating the convergence rate of sequences. An example of this is the well-known Aitken-δ^2 method, in which a new sequence is derived from the original sequence, which hopefully converges much faster than the original one.[9] Although many variants of this technique and many related algorithms exist, concentration below focuses on one particular method, commonly referred to as Wynn's ϵ algorithm. First, a brief discussion of the essential ingredients of vector-sequence extrapolation methods is presented, followed by the extension of these algorithms to the Euler equations as well as examples.

For the Aitken-δ^2 method, the sequence is derived by assuming that the original members of the sequence s_n can be adequately described as

$$s = s_n + \kappa\rho^n \tag{2.5.16}$$

where s is the limiting value of the sequence and κ and ρ are constants. By evaluating Eq. (2.5.16) at n, $n + 1$, and $n + 2$, the limiting value of the sequence may be obtained from the solution of the set of simultaneous equations for s, κ, and ρ. If the original sequence is accurately described by Eq. (2.5.16), then the exact answer will be obtained. If, on the other hand, Eq. (2.5.16) does not provide a precise description of s_n, then application of the procedure will be only approximate, but may still provide a better estimate for s than is currently available. This estimate for the value of s is given by

$$s_n^{(1)} = \frac{s_{n+2}^{(0)} s_n^{(0)} - \left(s_{n+1}^{(0)}\right)^2}{s_{n+2}^{(0)} + s_n^{(0)} - 2s_{n+1}^{(0)}} \tag{2.5.17}$$

where the superscript (0) is used to denote terms in the original sequence. This new sequence may now be used to define another sequence given by

$$s_n^{(2)} = \frac{s_{n+2}^{(1)} s_n^{(1)} - (s_{n+1}^{(1)})^2}{s_{n+2}^{(1)} + s_n^{(1)} - 2s_{n+1}^{(1)}} \tag{2.5.18}$$

which may supply a further improvement to the limiting value of the sequence. This procedure can be applied repeatedly, increasing the accuracy each time over the previous estimate.

An example of this procedure, borrowed from Ref. 9, is given below. The original sequence is taken to be the first nine terms in the series given by

$$\sum_{k=1}^{\infty} \frac{(-1)^{k+1}}{k} \tag{2.5.19}$$

which is the Taylor series expansion for $\ln(1+x)$ evaluated at $x = 1$.

In the current example, the solution is obtained to seven-digit accuracy ($\ln(2) = 0.6931472$) at the end of the extrapolation procedure with only the first nine partial sums. Note that to obtain similar accuracy from simply summing the series directly would require approximately 10 million terms.[9] The acceleration procedure is clearly very useful in this case.

A shortcoming of the above technique lies in the underlying assumption that the sequence behaves similarly to that described by Eq. (2.5.16). For this reason, the above extrapolation procedure is most effective on geometric series and becomes less effective as the series deviates from this behavior. To overcome this shortcoming, Shanks[10] derived other extrapolations based on the assumption that the sequence may be described in the more general form

$$s = s_n + a_1 \rho_1^n + \cdots + a_N \rho_N^n \tag{2.5.20}$$

which is referred to as the N^{th} order Shank's transformation. Equation (2.5.20) is evaluated for five values of n resulting in a set of equations that can be used to solve for s in much the same manner as to obtain Eq. (2.5.17). Although this higher order transformation may provide a more accurate representation of a general sequence, the implementation in this manner can be inefficient for higher order transforms.

In Ref. 11, Wynn describes an algorithm for computing the higher order extrapolations that is more efficient than the process described above. In this method, a table of values is constructed from the original sequence with the relationship

$$\epsilon_{k+1}^{(m)} = \epsilon_{k-1}^{(m+1)} + \left(\epsilon_k^{(m+1)} - \epsilon_k^{(m)}\right)^{-1} \tag{2.5.21}$$

where the values of $\epsilon^{(m)}$ are assumed to be arranged in a format shown in Fig. 2.5.7.

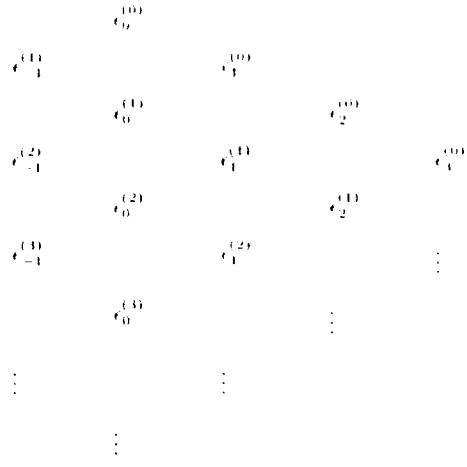


Figure 2.5.7 Format for ϵ algorithm.

The initial conditions are taken to be

$$\begin{aligned} \epsilon_{-1}^{(m)} &= 0, \qquad m = 1, 2, \ldots \\ \epsilon_0^{(m)} &= s_m, \qquad m = 0, 1, \ldots \end{aligned} \tag{2.5.22}$$

where s_m represents the terms of the original sequence. Note that the odd numbered columns do not represent actual approximations to the terms in the sequence, but are intermediate calculations necessary for the calculation of the even columns ϵ_{2m}^n $(m = 0, 1, \ldots)$.

n	s_n	$s_n^{(1)}$	$s_n^{(2)}$	$s_n^{(3)}$	$s_n^{(4)}$
1	1.0	0.7	0.6932773	0.6931489	0.6931472
2	0.5	0.6904762	0.6931058	0.6931467	
3	0.8333333	0.6944444	0.6931633	0.6931474	
4	0.5833333	0.6924242	0.6931399		
5	0.7833333	0.6935897	0.6931508		
6	0.6166667	0.6928571			
7	0.7595238	0.6933473			
8	0.6345238				
9	0.7456349				

Figure 2.5.6 Illustration of Aitken-δ^2 method.

In this procedure, each value of $\epsilon_{s+1}^{(m)}$ shown in Fig. 2.5.8 is calculated from Eq. (2.5.21).

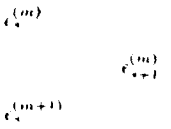


Figure 2.5.8 Module used for the computation of $\epsilon_{s+1}^{(m)}$.

In Ref. 12, a relation is given that allows the computation of the even columns without direct computation of the odd columns. This relationship is given by

$$E = C + \frac{(N - C)(S - C)}{(S - C) + (N - C) - \frac{\theta(N - C)(S - C)}{(W - C)}} \tag{2.5.23}$$

where

$$\begin{aligned} E &= \epsilon_{s+2}^{(m)} \quad W = \epsilon_{s-2}^{(m+2)} \quad C = \epsilon_{s}^{(m+1)} \\ N &= \epsilon_{s}^{(m)} \quad S = \epsilon_{s}^{(m+2)} \end{aligned} \tag{2.5.24}$$

Note that θ is zero for the first column and unity for subsequent columns; the continuous use of $\theta = 0$ corresponds to repeated first-order transformations. Although not shown, the use of the ϵ algorithm as described above to the series given in Eq. (2.5.19) yields results comparable to those shown in Fig. 2.5.6.

As previously mentioned, the effectiveness of the acceleration technique is strongly dependent on the assumption that the sequence behaves in the manner given by Eq. (2.5.20). Because only N terms are included, only eigenvectors associated with the first N dominant eigenvectors of the iteration scheme may be effectively eliminated. Therefore, an algorithm must be used that acts as a preconditioner, so that most of the eigenvalues of the iteration scheme are forced to be approximately equal. In this way, the number of terms contained in the original sequence may be kept as low as possible.

Application for the Euler Equations

To apply the above technique to systems of equations such as those that arise in Euler solvers, the ϵ algorithm has been generalized for systems by Hafez in Ref. 12. In this reference, a form similar to that in Eq. (2.5.23) is given by

$$\frac{\vec{E} - \vec{C}}{\delta^2} = \frac{\vec{N} - \vec{C}}{\alpha^2} + \frac{\vec{S} - \vec{C}}{\beta^2} - \frac{\vec{W} - \vec{C}}{\gamma^2} \cdot \theta \tag{2.5.25}$$

where

$$\delta^2 = \left[\left(\frac{\vec{N} - \vec{C}}{\alpha^2} + \frac{\vec{S} - \vec{C}}{\beta^2} - \theta \frac{\vec{W} - \vec{C}}{\gamma^2} \right)^2 \right]^{-1} \tag{2.5.26}$$

$$\begin{aligned} \alpha^2 &= \left\| \vec{N} - \vec{C} \right\| \\ \beta^2 &= \left\| \vec{S} - \vec{C} \right\| \\ \gamma^2 &= \left\| \vec{W} - \vec{C} \right\| \end{aligned} \tag{2.5.27}$$

and the inverse of a vector is defined by

$$\vec{X}^{-1} = \frac{\vec{X}}{\vec{X} \cdot \vec{X}} \tag{2.5.28}$$

In Figs. 2.5.9 and 2.5.10, examples are presented from Ref. 12 in which the above procedure is applied to the computation of a NACA 0012 airfoil at transonic conditions. For the computation, the explicit multigrid, multistage time-stepping scheme of Jameson is used, and only five terms are included in the initial sequence.

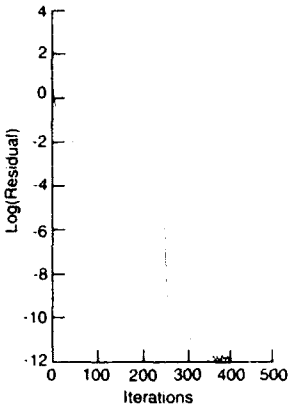


Figure 2.5.9 Effect of vector-sequence acceleration applied after 250 time steps for NACA 0012 airfoil; $M_\infty = 0.8$ and $\alpha = 1.25°$.

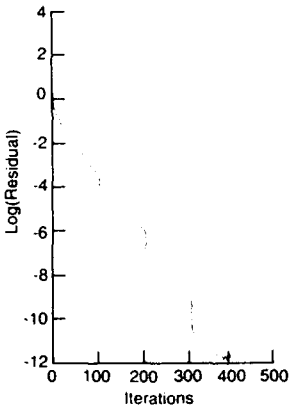


Figure 2.5.10 Effect of vector-sequence acceleration applied after 100, 200, and 300 time steps for NACA 0012 airfoil; $M_\infty = 0.8$ and $\alpha = 1.25°$.

Shown in Fig. 2.5.9 is the convergence history obtained by using the ϵ algorithm after 250 time steps. As seen, the residual drops dramatically at this point, which indicates the effectiveness of the acceleration procedure. Note, however, that at the point the acceleration is applied, the residual has been reduced by about 6 orders of magnitude, which should be more than sufficient for obtaining global quantities such as lift and drag. An attempt to apply the procedure earlier in the iteration history is shown in Fig. 2.5.10. As seen, the effectiveness of the current algorithm, when applied after 100 iterations, is minimal. After 200 iterations, however, a sudden drop in the residual is observed, and a further drop is seen at 300 iterations, where the algorithm is applied once again.

Although the ϵ algorithm can be very effective at achieving dramatic reductions in the residuals, systematic knowledge of when to apply the algorithm and how often is not clear. In the results shown above, for example, although the acceleration is impressive, at the point the method is applied and is effective, global quantities such as lift and drag are likely to be fully converged. Further research is required to gain more knowledge in this area. Furthermore, the implementation of this technique of convergence acceleration requires storage of twice the number of terms included in the initial sequence; for the example given above in which five terms in the initial sequence are used, the solution would need to be stored 10 times. For very large problems this requirement could become prohibitive.

2.5.4 Multigrid Acceleration

Introduction to Multigrid

One of the most successful and widely used methods of convergence acceleration is multigrid. The greatest benefit of this method is that the convergence rate (i.e., the spectral radius that measures the ratio of errors at successive time steps) remains constant, independent of the mesh spacing. In this way, solutions can be obtained in $\mathcal{O}(N)$ operations (i.e., the computational cost varies linearly with the total number of grid points N). Without convergence acceleration, the computational cost is considerably higher because of the penalty associated with a deterioration of the convergence rate as the mesh spacing decreases. Although most of the existing theory on multigrid methods pertains specifically to elliptic equations, a number of references (for example, 3, 13, 14, 15, and 16) have shown that the multigrid method can greatly accelerate the convergence rate of numerical schemes used to solve the Euler equations.

A brief description of the elements of multigrid is given below. The method is first described for the solution of a general nonlinear equation. The implementation of the full-approximation scheme (FAS) for the Euler equations is then discussed.

Description of Multigrid

The multigrid method most widely used for accelerating the convergence of iterative methods to solve the Euler equations is the full-approximation scheme (FAS) that appears in many references [17,18,19,20] and is summarized below. First, consider the solution of a general nonlinear system of equations

$$L(Q) = S \tag{2.5.29}$$

where L is a general nonlinear operator, Q is the solution vector of unknowns, and S represents a forcing function.

Equation (2.5.29) is solved numerically by dividing the domain into discrete cells that yield a system of equations to be solved simultaneously at each point as

$$L_N(Q_N) = S_N \tag{2.5.30}$$

where Q_N is the exact solution to the discretized system and L_N is the discrete analog of the operator L. If initial conditions are close enough to the final solution, Eq. (2.5.30) could be solved iteratively with Newton iteration. This approach, however, may be prohibitively expensive if the number of unknowns is large which typically occurs in multidimensional problems. Many other iterative schemes have, therefore, been devised that require significantly fewer operations. After a few iterations, however, these methods generally exhibit a slow convergence rate, which reduces the residuals by a very small amount each time.[18] The reason for the slow asymptotic convergence rate is the inadequate damping of the low-frequency errors.[21]

The multigrid method efficiently damps the low-frequency errors by using a sequence of grids $G_1, G_2, \ldots, G_N$, where G_N denotes the finest grid, from which successively coarser grids $G_{N-1}, G_{N-2}, \ldots, G_1$ are created. In a structured-grid full-coarsening algorithm, the coarser grids are constructed by deleting every other grid line in each coordinate direction. In practice, and particularly for unstructured grids, the coarser grids need not be constructed as a subset of the finest grid, (i.e., they can be created independently of the finest grid). In this context, the high-frequency error components on a given grid are those that cannot be resolved on the next coarser mesh because of the increased grid spacing. If an iterative method is chosen that quickly damps the high-frequency errors on a given grid, then after a few iterations, the remaining errors will be those associated with the smoother, low-frequency error components. Because these components appear as higher frequencies on coarser meshes, a sequence of coarser grids can be effectively used to accelerate the convergence rate on the finest grid. Therefore, the low-frequency errors on the fine grid that are usually responsible for slow convergence can be efficiently damped on the coarser grids. These computations are relatively inexpensive so the total overhead of the method is not excessively high. For example, the work required to solve the equations on all the grids in relation to that required to solve on just the finest grid can be estimated by

$$1 + \frac{1}{4} + \frac{1}{16} + \frac{1}{64} + \cdots \approx 1\frac{1}{3} \tag{2.5.31}$$

for two dimensions and

$$1 + \frac{1}{8} + \frac{1}{64} + \frac{1}{512} + \cdots \approx 1\frac{1}{7} \tag{2.5.32}$$

for three-dimensional calculations. Note that the above estimates are for structured grids from which coarser grids are formed by removing every other mesh line in all directions. Also, these estimates only account for the number of unknowns on the various grids and do not consider the extra residual computations necessary to compute the relative truncation error.

To use the coarser grids an equation must be obtained on the fine mesh that can be accurately represented by the coarser mesh. Neither the solution nor the high-frequency error components on the fine grid can generally be resolved on a coarser grid. The high-frequency errors, however, can be sufficiently damped on a fine grid by using iterative schemes specifically designed to damp high-frequency errors in the solution, so that the remaining errors will be composed of only low-frequency components that can be adequately represented on coarser meshes. Because only the low-frequency errors may be represented well on coarser meshes, it is necessary to obtain an equation on the fine mesh in terms of the errors.

To solve iteratively, Eq. (2.5.30) is solved approximately at each step as

$$L_N(q_N) = S_N + R_N \tag{2.5.33}$$

where q_N^c is the most current approximation to Q_N and R_N is the residual that will be zero only when $q_N^c = Q_N$. Hence, the exact discrete solution is obtained. Equation 2.5.33 is subtracted from Eq. (2.5.30) to yield an equation on the finest grid in terms of the residual:

$$L_N(Q_N) - L_N(q_N^c) = -R_N \tag{2.5.34}$$

With the assumption that the high-frequency errors have been previously smoothed, the fine-grid residual Eq. (2.5.34) can be adequately approximated on a coarser mesh as

$$L_{N-1}(Q_{N-1}) = \hat{I}_N^{N-1}(-R_N) + L_{N-1}\left(I_N^{N-1} q_N^c\right) \tag{2.5.35}$$

where I_N^{N-1} and $\hat{I}_N^{N-1}$ are restriction operators for transferring both the dependent variables and the residual from the fine grid to the coarse grid, respectively. Here, $I_N^{N-1} q_N^c$ serves as an *initial approximation to the solution on the coarse mesh*; Q_{N-1} is the exact solution of the coarse-grid problem and is the sum of the initial approximation and a correction.[20] Because the full solution is computed and stored on each grid level (as opposed to only the corrections, which is all that is required for a linear equation), this process is referred to as the FAS.

On a sufficiently coarse grid, Eq. (2.5.35) can be solved exactly with a variety of numerical techniques to obtain Q_{N-1}, from which the coarse-grid correction can be formed as

$$V_{N-1} = Q_{N-1} - I_N^{N-1} q_N^c \tag{2.5.36}$$

This can then be transferred to the fine grid and used as a correction to q_N^c, which is replaced by its previous value plus the prolongated correction

$$q_N^c \leftarrow q_N^c + \hat{I}_{N-1}^N V_{N-1} \tag{2.5.37}$$

This process yields a simple FAS two-level algorithm where the operations on the coarse grid (Eqs. 2.5.35)-(2.5.37) used to update the fine-grid solution are termed the coarse-grid correction. Normally, the exact solution of Eq. (2.5.35) can be expensive to obtain. Also, because the correction on the coarse grid serves only as an approximation to the fine-grid correction, the exact solution of Eq. (2.5.35) is *not required. Therefore, instead of solving* Eq. (2.5.35) to completion, several iterations can be carried out to get a reasonable approximation of Q_{N-1}. For an approximate solution q_{N-1}^c of Eq. (2.5.35), a corresponding coarse-grid residual R_{N-1} can be defined from

$$L_{N-1}(q_{N-1}^c) = \hat{I}_N^{N-1}(-R_N) + L_{N-1}\left(I_N^{N-1} q_N^c\right) + R_{N-1} \tag{2.5.38}$$

whose solution differs from the solution of Eq. (2.5.35) only by the residual term R_{N-1}, which will be zero when $q_{N-1}^c = Q_{N-1}$. If the errors are smooth, then subtraction of Eq. (2.5.38) from Eq. (2.5.35) yields an equation that can be well represented on a still coarser mesh G_{N-2}. If this equation is written on G_{N-2}, then

$$L_{N-2}(Q_{N-2}) = \hat{I}_{N-1}^{N-2}(-R_{N-1}) + L_{N-2}\left(I_{N-1}^{N-2} q_{N-1}^c\right) \tag{2.5.39}$$

where Eq. (2.5.38) is used to determine R_{N-1}. The solution may be obtained in one of three ways: by solving Eq. (2.5.39) exactly, by approximating by several iterations, or by introducing more coarse-grid levels. On all coarse grids, one or more FAS cycles (smoothing followed by coarse-grid correction) are completed. In this manner, each of the coarse meshes is used to obtain a correction for the solution on the next finest mesh. Because only the equations for smooth error components may be represented well on coarser grids, only the corrections (and not the full solution) must be passed from a coarse grid to the next finest grid.[18]

By using Eq. (2.5.33), note that Eq. (2.5.35) can be recast as

$$L_{N-1}(Q_{N-1}) = S_{N-1} + \tau_{N-1} = P_{N-1} \tag{2.5.40}$$

where

$$S_{N-1} = \hat{I}_N^{N-1} S_N \tag{2.5.41}$$

$$\tau_{N-1} = L_{N-1}\left(I_N^{N-1} q_N^c\right) - \hat{I}_N^{N-1}\left(L_N(q_N^c)\right) \tag{2.5.42}$$

Here, τ_{N-1} is the relative truncation error (or defect correction) between the grids, so that the solution on the coarse grid is driven by the fine grid, and the defect correction accounts for the difference in the truncation error between the coarse and fine grids.[18] The analogous equation for grid G_{N-2} is given by

$$L_{N-2}(Q_{N-2}) = S_{N-2} + \tau_{N-2} \tag{2.5.43}$$

$$S_{N-2} = \hat{I}_{N-1}^{N-2} S_{N-1} \tag{2.5.44}$$

and

$$\tau_{N-2} = L_{N-2}\left(I_{N-1}^{N-2} q_{N-1}^c\right) - \hat{I}_{N-1}^{N-2}\left[L_{N-1}(q_{N-1}^c)\right] + \hat{I}_{N-1}^{N-2} \tau_{N-1} \tag{2.5.45}$$

Note that the relative truncation error on the $N-2$ grid is the sum of the relative truncation error between grids N and $N-1$, as well as $N-1$ and $N-2$. Thus, the equations solved on the coarser meshes (Eqs. (2.5.40) and (2.5.43)), for example, appear exactly as the original equation, except that a forcing function appears on the coarser meshes. The result is that the coarse meshes can be updated with the same scheme that is used on the fine mesh, with only a slight modification to the right-hand side.

Algorithm for Euler Equations

For the steady Euler equations that are written in generalized coordinates, Eq. (2.5.30) can be written as

$$L_N(Q_N) = \delta_\xi \hat{F} + \delta_\eta \hat{G} + \delta_\zeta \hat{H} = 0 \tag{2.5.46}$$

In the multigrid solution process, a forcing function arises on the coarse grids from restricting the residual equation on a fine mesh to the coarser mesh. The resulting equation, to be solved on any mesh G_i, can be written as

$$L_i(Q_i) = \tau_i \tag{2.5.47}$$

where τ_i is equal to 0 on the finest mesh and is the relative truncation error on each of the coarser grids. The solution of Eq. (2.5.47) is generally updated by introducing a time derivative of the dependent variables to the left-hand side so that the solution can be advanced in time with methods such

as Runge-Kutta, approximate factorization, or relaxation methods previously described. For implicit methods, the resulting scheme, written on mesh G_i, is given by

$$N\Delta q_i = -\Delta t[L_i(q_i) - \tau_i] = -\Delta t R_i \qquad (2.5.48)$$

where N is the implicit operator of the considered scheme and $L_i(q_i)$ on the right-hand side results from the linearization of $L_i(Q_i)$ from the backward Euler time integration. Note, as mentioned previously, that even on the coarse meshes, where τ_i is nonzero, Eq. (2.5.48) maintains the same form as the equation on the fine mesh.

Several strategies exist for deciding when to switch from one grid level to another; these generally fall under the categories of fixed- or adaptive-cycle algorithms. The strategy most commonly used is a fixed-cycle strategy, in which each global cycle consists of a set number of FAS cycles on each of the coarser grids. Recall that one FAS cycle on any grid consists of a smoothing step, followed by a coarse-grid correction. A predetermined number of iterations are performed on each grid level to smooth the errors.

The conserved variables are transferred to successively coarser grids each time by the rule

$$Q_{i+1} = I_i^{i+1} Q_i \qquad (2.5.49)$$

where I_i^{i+1} is a volume-weighted restriction operator that transfers values on the fine grid to the coarser grid and is defined by

$$I_i^{i+1} Q_i = \frac{\sum VQ}{\sum V} \qquad (2.5.50)$$

The summations are taken over all the fine-grid cells that make up the coarse-grid cell. Restriction of the dependent variables in this manner conserves the total mass, momentum, and energy of the fine grid on the coarser grids. In general, the relative truncation error is calculated on the coarse grid as

$$\tau_{i+1} = L_{i+1}\left(I_i^{i+1} q_i\right) - \tilde{I}_i^{i+1} R \qquad (2.5.51)$$

where $\tilde{I}_i^{i+1}$ is the restriction operator for the residual, generally defined as

$$I_i^{i+1} R_i = \sum R_i \qquad (2.5.52)$$

where, again, the summation is over the cells on the fine grid that make up the coarse-grid cell. By summing the residuals, the surface integral of the fluxes that cross the cell boundaries on the coarse grid is the same as would occur by integrating around the fine-grid cells that make up the coarse grid. Several time steps can be conducted with an iterative scheme to get an approximation to the steady solution on G_{i+1}, with the right-hand side modified to include the relative truncation error. If only one coarse grid is used to correct the finest grid, the result is the simple FAS two-level cycle. On the other hand, if more grid levels are introduced so that one or more FAS cycles can be recursively carried out on each subsequent coarse-grid level to better approximate Q_{i+1}, then a multilevel algorithm results.

When only one FAS cycle is carried out for each of the coarser grids, the resulting global cycling strategy is termed a V-cycle, which is depicted in Fig. 2.5.11. Another cycling strategy of interest, which is shown in Fig. 2.5.12, is termed a W-cycle and results when two FAS cycles are used on each of the coarser meshes. Results will be shown in the next section for both types of cycles. The corrections on coarse meshes are prolongated to the next finest mesh with trilinear interpolation and no additional iteration steps between meshes. When a W-cycle is used, however, note that an iteration is carried out at the beginning of each FAS cycle correction to smooth the high frequencies.

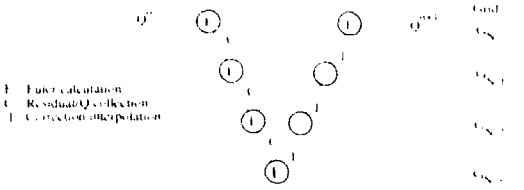


Figure 2.5.11 Multigrid V-cycle.

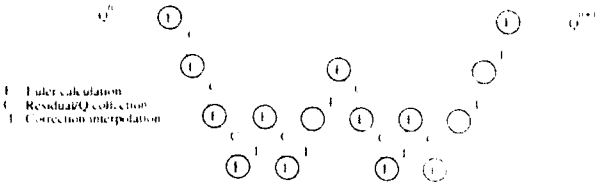


Figure 2.5.12 Multigrid W-cycle.

To further clarify the multigrid procedure, the overall process is summarized as follows for an exemplary case, where three grid levels are used in a V-cycle:

1. Start on the finest grid and smooth the errors by completing one iteration of Eq. (2.5.48) with $\tau_i = 0$.

2. Calculate the residual on the fine grid from Eq. (2.5.33), where $L_N(q_N)$ is given on the right-hand side of Eq. (2.5.46) and $S_N = 0$.

3. Restrict the dependent variables to the first coarse grid G_{N-1} by using Eq. (2.5.49).

4. Restrict the residual from the finest grid to G_{N-1} with Eq. (2.5.52) and calculate the relative truncation error with Eq. (2.5.51).

5. Calculate the right-hand side of Eq. (2.5.48) and update the solution on mesh G_{N-1}. (This smooths the errors on this grid so that a coarser grid can be introduced.)

6. Calculate the residual on this mesh with Eq. (2.5.38). Note that this can be written as

$$R_{N-1} = L_{N-1}(q_{N-1}) - \tau_{N-1} \qquad (2.5.53)$$

Because τ_{N-1} has been previously calculated, the residual is easily calculated by simply calculating $L_{N-1}(q_{N-1})$ from the most current values of the dependent variables on the mesh and then subtracting τ_{N-1}.

7. Restrict the dependent variables on G_{N-1} to G_{N-2} by using Eq. (2.5.49).

8. Restrict the residual from Eq. (2.5.53) to the $N-2$ grid and calculate τ_{N-2} from Eq. (2.5.51).

9. Calculate the right-hand side of Eq. (2.5.48) and update the solution on this mesh. Because this is the coarsest mesh used in the present example, three iterations of Eq. (2.5.48) are used to get an approximation to Q_{N-2}. During each step, the right-hand side is updated to use the most current values of the dependent variables in $L_{N-2}(q_{N-2})$. Note that τ_{N-2} will not change.

10. Calculate the correction on this mesh to give

$$V_{N-2} = q_{N-2} - I_{N-2}^{N-1} q_{N-1} \tag{2.5.54}$$

11. Pass the correction to the next finest mesh with trilinear interpolation and update the solution to give

$$q_{N-1} = q_{N-1} + I_{N-2}^{N-1} V_{N-2} \tag{2.5.55}$$

Note that steps 5 through 11 make up one FAS cycle on grid $N-1$, in which steps 6 to 11 constitute a coarse-grid correction. At this point, if a W-cycle was employed, another FAS cycle (steps 5 to 11) would be repeated to update q_{N-1} further.

12. Calculate the correction on the $N-1$ mesh as

$$V_{N-1} = q_{N-1} - I_N^{N-1} q_N \tag{2.5.56}$$

13. Pass this correction to the finest mesh and update the solution to give

$$q_N = q_N + I_{N-1}^N V_{N-1} \tag{2.5.57}$$

14. Perform one smoothing iteration with Eq. (2.5.48) to smooth the errors.

Smoothing Algorithms

Many algorithms can be used to smooth the high-frequency errors. Both explicit and implicit algorithms have been used with success. For multigrid to succeed, the high-frequency errors must be effectively damped. The effectiveness of a scheme can be estimated by determining the smoothing factor of the algorithm with a Fourier analysis. The smoothing factor is defined as the maximum eigenvalue of the algorithm for frequencies greater than $\pi/2$ and less than $3\pi/2$. These frequencies represent those on a fine grid that are not resolvable on a coarser grid.

Examples of explicit algorithms that have been used successfully include the pioneering work done by Jameson,[22,23,3,16] in which a multistage Runge-Kutta scheme was used along with implicit residual smoothing. In addition, the coefficients of the Runge-Kutta algorithm have been chosen so that the damping of the high-frequency errors is enhanced. Various researchers who have used implicit algorithms including Anderson,[13] Mulder,[15] Yoon,[24] and Spekreijse,[25]

Examples of Multigrid Applications

An example of the application of multigrid for the Euler equations is shown below for a three-dimensional transonic flow computation over the ONERA M6 wing.[13] Comparisons are made with experimental data at a Reynolds number of 11.7 million,[26] which corresponds to conditions for which viscous effects are relatively small. The wing consists of symmetrical airfoil sections with a planform swept at 30° along the leading edge, an aspect ratio of 3.8, and a taper ratio of 0.56. A solution is obtained on a C-H mesh, which has a C type of mesh topology around the airfoil profile and an H type mesh in the spanwise direction.

The effectiveness of multigrid acceleration is demonstrated for a computation at transonic conditions with a Mach number of 0.84 and an angle of attack of 3.06°. Figures 2.5.13 and 2.5.14 show the effect of using multigrid on the residual and lift-coefficient histories for a mesh with over 210,000 points. As previously mentioned, the mesh is a $193 \times 33 \times 33$ C-H mesh that has 193 points along the airfoil and wake (110 of which are on the airfoil), 33 points approximately normal to the airfoil, and 33 points in the spanwise direction (17 of which are on the wing planform). For this calculation, an implicit, upwind differencing method is used for smoothing the errors, with a V-cycle and four grid levels (one fine and three coarser grids). The multigrid method is very effective in accelerating convergence of both the residual and lift coefficients. The residual is reduced to machine zero in 400 cycles, whereas the single-grid method has reduced the residual only between 1 and 2 orders of magnitude. The benefit of multigrid is especially pronounced in the lift-coefficient history where the lift-coefficient value is obtained to within 0.1 percent of its final value in 41 cycles. This is a dramatic improvement over the single-grid result, which required more than 400 cycles to converge to the same level of accuracy for the lift coefficient.

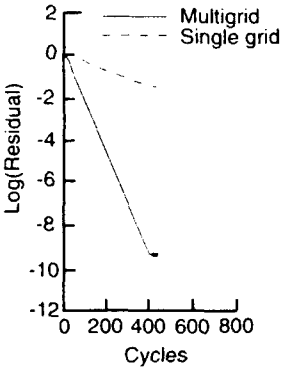


Figure 2.5.13 Effect of multigrid on residual history for ONERA M6 wing with $M_\infty = 0.84$ and $\alpha = 3.06°$.

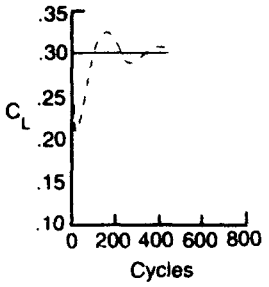


Figure 2.5.14 Effect of multigrid on lift history for ONERA M6 wing with $M_\infty = 0.84$ and $\alpha = 3.06°$.

The multigrid procedure has also been implemented into multiblock versions of several codes to handle complex geometries.[27,28,29,30,31] In addition, several examples of the application of multigrid to reduce the computational times required for time-accurate calculations can be found in Refs. 32, 33, 34, and 35.

2.5.5 GMRES

The generalized minimal residual (GMRES)[36] algorithm for solving a nonsymmetric linear system of equations has been extended to nonlinear problems and applied to Euler calculations by Wigton et al.[37] In this implementation, the equation considered for solution is written as

$$\mathbf{R}(\mathbf{w}) = 0 \tag{2.5.58}$$

where, for the Euler equations, $\mathbf{R}(\mathbf{w}) = 0$ represents the steady-state residual. The differential of $\mathbf{R}(\mathbf{w}) = 0$ in a general direction $\mathbf{p}$ is denoted by $\tilde{\mathbf{R}}(\mathbf{w};\mathbf{p})$ and is given by

$$\tilde{\mathbf{R}}(\mathbf{w};\mathbf{p}) = \lim_{\varepsilon \to 0} \frac{\mathbf{R}(\mathbf{w}+\varepsilon\mathbf{p}) - \mathbf{R}(\mathbf{w})}{\varepsilon} \tag{2.5.59}$$

Analogous to the procedure for linear systems, the GMRES algorithm first obtains k orthonormal search directions $\mathbf{p}_1, \mathbf{p}_2, \ldots, \mathbf{p}_k$ and then updates the solution as

$$\mathbf{w}^{n+1} = \mathbf{w}^n + \sum_{j=1}^{k} a_j \mathbf{p}_j \tag{2.5.60}$$

where the a_j are chosen to minimize

$$\begin{aligned} \|\mathbf{R}(\mathbf{w}^{n+1})\|^2 &= \left\| \mathbf{R}\left(\mathbf{w}^n + \sum_{j=1}^{k} a_j \mathbf{p}_j\right) \right\|^2 \\ &\approx \left\| \mathbf{R}(\mathbf{w}^n) + \sum_{j=1}^{k} a_j \tilde{\mathbf{R}}(\mathbf{w}^n;\mathbf{p}_j) \right\|^2 \end{aligned} \tag{2.5.61}$$

The orthogonal search directions are determined by a Gram-Schmidt process:

$$\mathbf{p}_1 = \frac{\mathbf{R}(\mathbf{w}^n)}{\|\mathbf{R}(\mathbf{w}^n)\|} \tag{2.5.62}$$

For $j = 1, 2, \ldots, k-1$,

$$\mathbf{p}_{j+1} = \tilde{\mathbf{R}}(\mathbf{w}^n;\mathbf{p}_j) - \sum_{i=1}^{j} b_{ij} \mathbf{p}_i \tag{2.5.63}$$

$$\mathbf{p}_{j+1} = \frac{\mathbf{p}_{j+1}}{\|\mathbf{p}_{j+1}\|} \tag{2.5.64}$$

where b_{ij} is the projection of $\tilde{\mathbf{R}}(\mathbf{w}^n;\mathbf{p}_j)$ in the direction of $\mathbf{p}_i$:

$$b_{ij} = \left[\tilde{\mathbf{R}}(\mathbf{w}^n;\mathbf{p}_j), \mathbf{p}_i\right] \tag{2.5.65}$$

In practice, the above process is actually applied to a "preconditioned" equation that has the same solution as the original problem, but has a more favorable distribution of eigenvalues. For the problem $\mathbf{R}(\mathbf{w}) = 0$, most computer codes generate an improved approximation to the current estimate of the solution as

$$\mathbf{w}^{n+1} = \mathbf{M}(\mathbf{w}^n) \tag{2.5.66}$$

where $\mathbf{M}$ represents some methodology such as line relaxation, approximate factorization, or Runge-Kutta time stepping. Convergence is achieved when $\mathbf{w}^{n+1} = \mathbf{w}^n$. Therefore, the solution of Eq. (2.5.58) can be replaced by the equation

$$\mathbf{R}'(\mathbf{w}) = \mathbf{w} - \mathbf{M}(\mathbf{w}) \tag{2.5.67}$$

for which GMRES is much more effective. Note, however, that every evaluation of $\mathbf{R}'(\mathbf{w})$ involves an evaluation of $\mathbf{M}$.

An example of results with GMRES to accelerate the convergence of an existing flow solver for a transonic calculation is shown in Fig. 2.5.15. Here, GMRES is applied to a two-dimensional central-differenced implicit Euler code denoted as ARC2D.[38] As seen in the figure, the use of GMRES can result in a significant increase in the convergence rate. After 400 function calls (where one function call is one evaluation of Eq. (2.5.67)), the residual is reduced about 4 orders of magnitude over that without GMRES.

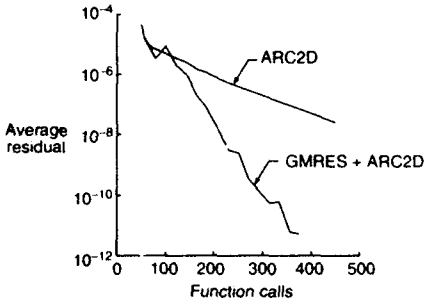


Figure 2.5.15 Convergence acceleration with GMRES.

2.5.6 Preconditioning

Recent work has been undertaken to accelerate the convergence rate of iterative schemes by essentially multiplying the time derivative by a matrix that allows faster convergence, but does not alter the steady state. The motivation for this is easily seen by examining the one-dimensional Euler equations

$$\frac{\partial \mathbf{w}}{\partial t} + \frac{\partial \mathbf{F}}{\partial x} = 0 \tag{2.5.68}$$

After this equation is linearized and a similarity transformation is used, this equation can be recast into the form

$$\frac{\partial \mathbf{q}}{\partial t} + \mathbf{\Lambda} \frac{\partial \mathbf{q}}{\partial x} = 0 \tag{2.5.69}$$

where $\mathbf{A}$ is a diagonal matrix whose entries are the eigenvalues of the flux Jacobian

$$\Lambda = \begin{bmatrix} u & 0 & 0 \\ 0 & u+a & 0 \\ 0 & 0 & u-a \end{bmatrix} \tag{2.5.70}$$

The equations given by Eq. (2.5.69) are now uncoupled, so that each equation can be approximated separately. For example, simple explicit time differencing can be used in conjunction with first-order spatial accuracy, where each equation is differenced according to the sign of the eigenvalue. The allowable time step for stability depends on the size of the maximum eigenvalue as well as on the grid spacing through the CFL number (defined as the product of the convective speed and the time step divided by the grid spacing). If a CFL is maintained less than unity for the simple explicit scheme considered, then the numerical characteristics completely enclose the physical ones. If all equations are advanced in time with the same Δt, then the CFL number for the equation whose convective speed (eigenvalue) is smallest may not be advanced nearly as fast as the stability criteria allows. For example, for $u = 0.5$ and $a = 1$, the limiting condition corresponds to the largest eigenvalue, which is $u + a = 1.5$. This corresponds to determining the time step by the speed of an acoustic wave that is moving to the right with a speed $u + a$. Note, however, that choosing the time step based on this eigenvalue means that the first equation (associated with the eigenvalue $\lambda_1 = u$) is advanced at a time step somewhat lower than the stability criteria requires. For the conditions chosen above, this restriction is not too prohibitive. However, for a low-speed flow, the wide disparity in the size of the eigenvalues can lead to slow convergence unless a time step is used separately for each equation, based on the individual eigenvalues.

The basic premise of preconditioning is to advance each equation with an optimum time step for each. For the one-dimensional case, this can be easily achieved by multiplying the right-hand side of Eq. (2.5.68) by a matrix, so that when the equation is diagonalized, all eigenvalues are equal:

$$\frac{\partial \mathbf{w}}{\partial t} + \mathbf{P}\frac{\partial \mathbf{F}}{\partial x} = 0 \tag{2.5.71}$$

Note that the preconditioning matrix does not change the steady state. Also note that $\mathbf{P}$ should be a positive definite matrix. Otherwise, the nature of the flow could be changed, as it would if $\mathbf{P} = -\mathbf{I}$, which would correspond to marching backward in time. For the one-dimensional case, this matrix is given by

$$\mathbf{P} = \left|\mathbf{A}^{-1}\right| = \left|\mathbf{T}\left(\frac{1}{\Lambda}\right)\mathbf{T}^{-1}\right| \tag{2.5.72}$$

where $\mathbf{T}$ and $\mathbf{T}^{-1}$ are the right and left eigenvectors of the matrix $\mathbf{A} = \partial\mathbf{F}/\partial\mathbf{w}$ and $1/\mathbf{A}$ is a diagonal matrix whose entries are the inverse of the eigenvalues of $\mathbf{A}$.

In the above example the criteria used in determining the optimum time step for each equation is based solely on taking the largest allowable time step for each equation. Other criteria can be used, such as selection of a time step to provide maximum damping of certain frequencies for use in a multigrid algorithm. Also note that if all the eigenvalues are of comparable size, such as for hypersonic flow, then no significant benefit would be expected.

The simplicity of the matrix in Eq. (2.5.72) is attributable to the fact that the one-dimensional Euler equations are easily diagonalized. The complexity of devising a preconditioner arises in multidimensions because the equations cannot be simultaneously diagonalized (with the exception of supersonic flow). However, recent work at preconditioning the equations in multidimensions has been undertaken with some success.[39,40,41,42,43]

In the work in Ref. 42, the preconditioning matrix is devised by first transforming the conservative dependent variables to those that yield a symmetric form for the linearized equations. The equations are then rotated into a coordinate system that is aligned with the flow direction; the resulting equations are given by

$$\frac{\partial \tilde{\mathbf{U}}}{\partial t} + \tilde{\mathbf{A}}\frac{\partial \tilde{\mathbf{U}}}{\partial s} + \tilde{\mathbf{B}}\frac{\partial \tilde{\mathbf{U}}}{\partial n} = 0 \tag{2.5.73}$$

where

$$\tilde{\mathbf{A}} = \begin{bmatrix} q & a & 0 & 0 \\ a & q & 0 & 0 \\ 0 & 0 & q & 0 \\ 0 & 0 & 0 & q \end{bmatrix} \tag{2.5.74}$$

$$\tilde{\mathbf{B}} = \begin{bmatrix} 0 & 0 & a & 0 \\ 0 & 0 & 0 & 0 \\ a & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \tag{2.5.75}$$

$$d\tilde{\mathbf{U}} = \begin{Bmatrix} dp/\rho a \\ du \\ dv \\ dp - a^2 d\rho \end{Bmatrix} \tag{2.5.76}$$

and $q = u\cos\theta + v\sin\theta$ is the magnitude of the velocity in the streamwise direction.

Considering the case of supersonic flow first and multiplying the residual terms in Eq. (2.5.73) by $\left|\tilde{\mathbf{A}}^{-1}\right|$ (equal to $\tilde{\mathbf{A}}^{-1}$ for supersonic flow) yields

$$\frac{\partial \tilde{\mathbf{U}}}{\partial t} = -\left(\mathbf{I}\frac{\partial \tilde{\mathbf{U}}}{\partial s} + \tilde{\mathbf{A}}^{-1}\tilde{\mathbf{B}}\frac{\partial \tilde{\mathbf{U}}}{\partial n}\right) \tag{2.5.77}$$

Because the flow is assumed to be supersonic, this equation can be simultaneously diagonalized with a transformation $d\tilde{\mathbf{V}} = \mathbf{T}^{-1}d\tilde{\mathbf{U}}$ to yield

$$\frac{\partial \tilde{\mathbf{V}}}{\partial t} = -\left(\mathbf{I}\frac{\partial \tilde{\mathbf{V}}}{\partial s} + \mathbf{\Lambda}\frac{\partial \tilde{\mathbf{V}}}{\partial n}\right) \tag{2.5.78}$$

Here, $\mathbf{A}$ is the diagonal matrix of the eigenvalues of $\tilde{\mathbf{A}}^{-1}\tilde{\mathbf{B}}$, and $\mathbf{T}$, $\mathbf{T}^{-1}$ are diagonalizing matrices. As discussed in Ref. 42, the wave speeds in the streamwise direction have been equalized. Unfortunately, the disparity in the acoustic wave speed in the direction normal to the streamlines is amplified; the ratio of the acoustic wave speeds to the convective wave speed is $M/\sqrt{M^2-1}$.

To make the acoustic and convective waves speeds equal, the ratio $\sqrt{M^2-1}/M$ is used to scale the acoustic waves by multiplying the right-hand side of Eq. (2.5.78) by the matrix X, which is defined as

$$X = \begin{bmatrix} \tau & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & \tau \end{bmatrix} \tag{2.5.79}$$

where $\tau = \sqrt{M^2 - 1}/M$. After converting back to symmetry variables, the preconditioned system of equations in streamlined coordinates, which is valid for supersonic flow, is given by

$$\frac{\partial \tilde{\mathbf{U}}}{\partial t} = -\mathbf{P}\left(\tilde{\mathbf{A}}\frac{\partial \tilde{\mathbf{U}}}{\partial s} + \tilde{\mathbf{B}}\frac{\partial \tilde{\mathbf{U}}}{\partial n}\right) \tag{2.5.80}$$

where

$$\mathbf{P} = \mathbf{T}\mathbf{X}\mathbf{T}^{-1}\mathbf{A}^{-1} = \begin{bmatrix} \frac{\tau}{\beta^2}M^2 & -\frac{\tau}{\beta^2}M & 0 & 0 \\ -\frac{\tau}{\beta^2}M & \frac{\tau}{\beta^2}+1 & 0 & 0 \\ 0 & 0 & \tau & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.5.81}$$

$\beta = \sqrt{M^2 - 1}$, and $\tau = \sqrt{1 - M^{-2}}$.

Difficulties arise for subsonic flow because the equations are no longer able to be simultaneously diagonalized. By assuming that the preconditioning matrix has a structure similar to Eq. (2.5.81), a preconditioning matrix can be obtained by requiring the convective waves be unchanged and that the acoustic waves travel at the flow velocity in the limit of zero Mach number. The final preconditioning matrix in the stream-aligned coordinate system in symmetry variables is identical to Eq. (2.5.81) except for subsonic flow:

$$\beta = \tau = \sqrt{1 - M^2} \qquad M < 1 \tag{2.5.82}$$

Recall that the preconditioning matrix given by Eq. (2.5.81) is for the stream-aligned coordinate system and symmetry variables; this matrix must be transformed back into the variables that will be solved numerically. For example, if the dependent variables are the conservative variables, the final preconditioning matrix, which simply multiplies the steady-state residual, is given by

$$\tilde{\mathbf{P}} = \frac{\partial \mathbf{w}}{\partial \mathbf{q}}\frac{\partial \mathbf{q}}{\partial \tilde{\mathbf{U}}}\frac{\partial \mathbf{U}}{\partial \tilde{\mathbf{U}}}\mathbf{P}\frac{\partial \tilde{\mathbf{U}}}{\partial \mathbf{U}}\frac{\partial \mathbf{U}}{\partial \mathbf{q}}\frac{\partial \mathbf{q}}{\partial \mathbf{w}} \tag{2.5.83}$$

where $\partial \mathbf{w}/\partial \mathbf{q}$ is the Jacobian matrix for transforming from primitive to conservative variables, $\partial \mathbf{q}/\partial \mathbf{U}$ transforms the symmetry variables to primitive ones, and $\partial \mathbf{U}/\partial \tilde{\mathbf{U}}$ relates the symmetry variables in the stream-aligned coordinate system to those in a Cartesian system:

$$\frac{\partial \mathbf{w}}{\partial \mathbf{q}} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ u & \rho & 0 & 0 \\ v & 0 & \rho & 0 \\ \frac{1}{2}(u^2+v^2) & \rho u & \rho v & \frac{1}{\gamma-1} \end{bmatrix} \tag{2.5.84}$$

$$\frac{\partial \mathbf{q}}{\partial \tilde{\mathbf{U}}} = \begin{bmatrix} \rho/a & 0 & 0 & -1/a^2 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ \rho a & 0 & 0 & 0 \end{bmatrix} \tag{2.5.85}$$

$$\frac{\partial \mathbf{U}}{\partial \tilde{\mathbf{U}}} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta & 0 \\ 0 & \sin\theta & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.5.86}$$

To devise flux formulas for computing the steady-state residual, the artificial viscosity must be modified for certain formulations. In particular, for Roe's approximate Riemann solver,[44] Refs. 42 and 43 show that the dissipation matrix must be modified in order to maintain the full benefits of preconditioning, both in stability and in accuracy.

An example from Ref. 42 is shown in Fig. 2.5.16, in which the preconditioning discussed above is applied to obtain the flow field around a NACA 0012 airfoil at a low Mach number (0.05). The numerical scheme used for this calculation is an explicit two-stage Runge-Kutta scheme, which uses Roe's approximate Riemann solver for discretization of the residual. The preconditioning dramatically decreases the number of iterations required to obtain a fully converged solution.

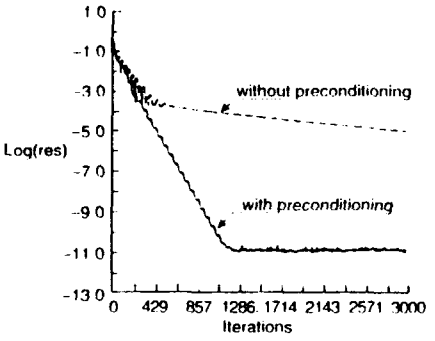


Figure 2.5.16 Convergence history with and without matrix preconditioning from Ref. 42.

2.5.7 Enthalpy Damping

The last method to be discussed for accelerating the convergence of numerical schemes to solve the Euler equations requires a modification of the governing equations that does not alter the steady-state solution. For a steady-state solution, because the total enthalpy is constant, a term proportional to the difference $H - H_\infty$ may be used as a forcing function to accelerate convergence. This technique has been introduced in Ref. 45 and is referred to as enthalpy damping.

A source term is simply added so that the governing equations are given by

$$\begin{aligned} &\frac{\partial \rho}{\partial t} + \frac{\partial \rho u}{\partial x} + \frac{\partial \rho v}{\partial y} + \alpha\rho(H - H_\infty) = 0 \\ &\frac{\partial \rho u}{\partial t} + \frac{\partial(\rho u^2 + p)}{\partial x} + \frac{\partial \rho uv}{\partial y} + \alpha\rho u(H - H_\infty) = 0 \\ &\frac{\partial \rho v}{\partial t} + \frac{\partial \rho uv}{\partial x} + \frac{\partial(\rho v^2 + p)}{\partial y} + \alpha\rho v(H - H_\infty) = 0 \\ &\frac{\partial \rho E}{\partial t} + \frac{\partial \rho uH}{\partial x} + \frac{\partial \rho vH}{\partial y} + \alpha\rho H(H - H_\infty) = 0 \end{aligned} \tag{2.5.87}$$

where a typical value of α is 0.25. Because H is equal to H_∞ at convergence the steady-state solution is not altered. Note that this technique is only applicable for flows with a constant total enthalpy and, therefore, may not be used for some simulations where propulsion effects are accounted for through specification of variations in total enthalpy.

2.5.8 References

1. LI, C. P., "Numerical Solution of Viscous Reacting Blunt Body Flows of a Multicomponent Mixture," AIAA 73–202, 1973.
2. LERAT, A., "Une Class de Schemes aux Differences Implicites pour les Systems Hyperboliques de Lois de Conservation," Compte Rendus Acad. Sciences Paris, vol. 288A, 1979.
3. JAMESON, A., "Solution of the Euler Equations for Two Dimensional Transonic Flow by a Multigrid Method," MAE Report Princeton University, June 1983.
4. BLAZEK, J., KROLL, N., RADESPIEL, R., and ROSSOW, C., "Upwind Implicit Residual Smoothing Method for Multi-Stage Schemes," AIAA 91–1533–CP, June 1991.
5. VENKATAKRISHNAN, V., *Computation of Unsteady Transonic Flows Over Moving Airfoils*. PhD thesis, Princeton University, 1986.
6. FRINK, N. T., PARIKH, P., and PIRZADEH, S., "A Fast Upwind Solver for the Euler Equations on Three-Dimensional Unstructured Meshes," AIAA 91–0102, Jan. 1991.
7. BLAZEK, J., KROLL, N., and ROSSOW, C. C., "A Comparison of Several Implicit Smoothing Methods," ICFD Conference on Numerical Methods for Fluid Dynamics, Reading, U.K, 1992.
8. BLAZEK, J., ROSSOW, C. C., KROLL, N., and SWANSON, R. C., "A Comparison of Several Implicit Residual Smoothing Methods in Combination with Multigrid," 13th International Conference on Numerical Methods for Fluid Dynamics, Rome, July 1992.
9. GREENBERG, M. D., *Foundations of Applied Mathematics*. Prentice Hall, 1978.
10. SHANKS, D., "Non-linear Transformation of Divergent and Slowly Convergent Sequences," Journal of Mathematics and Physics, vol. XXXIV, pp. 1–42, 1955.
11. WYNN, P., "On the Convergence and Stability of the Epsilon Algorithm," J. Siam Num. Anal., vol. 3, no. 1, pp. 91–122, 1966.
12. HAFEZ, M., PALANISWAMY, S., KURUVILA, G., and SALAS, M., "Applications of Wynn's Epsilon Algorithm to Transonic Flow Calculations," AIAA 87–1143, 1987.
13. ANDERSON, W. K., THOMAS, J. L., and WHITFIELD, D. L., "Three-Dimensional Multigrid Algorithms for the Flux-Split Euler Equations," Tech. Rep. 2829, NASA, 1988.
14. NI, R.-H., "A Multiple-Grid Scheme for Solving the Euler Equations," AIAA Journal, vol. 20, Nov. 1982.
15. MULDER, W., "Multigrid Relaxation for the Euler Equations," J. of Comp. Phys., vol. 60, pp. 235–252, Sept. 1985.
16. JAMESON, A., and BAKER, T., "Multigrid Solution of the Euler Equations for Aircraft Configurations," AIAA 84–0093, Jan. 1984.
17. HACKBUSH, W., *Multi-Grid Methods and Applications*. Springer-Verlag, 1985.
18. BRANDT, A., "Multi-Level Adaptive Solutions to Boundary-Value Problems," Math. Comput., vol. 31, pp. 333–390, Apr. 1977.
19. BRANDT, A., "Guide to Multigrid Development," in *Multigrid Methods: Volume 960 of Lecture Notes in Mathematics*, pp. 220–312, Springer Verlag, 1982.
20. BRANDT, A., "Multilevel Adaptive Computations in Fluid Dynamics," AIAA, vol. 18, pp. 1165–1172, Oct. 1980.
21. STUBEN, K., and TROTTENBERG, U., "Multigrid Methods: Fundamental Algorithms, Model Problems Analysis and Applications," in *Multigrid Methods: Volume 960 of Lecture Notes in Mathematics*, pp. 1–176, Springer Verlag, 1982.
22. JAMESON, A., "Solution of the Euler Equations by a Multigrid Method," Applied Mathematics and Computations, vol. 13, pp. 345–356, 1983.
23. JAMESON, A., "Multigrid Algorithms for Compressible Flow Calculations," Lecture Notes in Mathematics, Proceedings of the 2nd European Conference on Multigrid Methods, Cologne, vol. 1228, pp. 166–201, 1985.
24. JAMESON, A., and YOON, S., "LU Implicit Schemes with Multiple Grids for the Euler Equations," AIAA 86–0105, 1986.
25. SPEKREIJSE, S., "Multigrid Solution of the Steady Euler Equations," CWI-tract 46 (Center for Mathematics and Computer Science, Amsterdam), 1988.
26. SCHMITT, V., and CHARPIN, F., "Pressure Distributions on the ONERA-M6 Wing at Transonic Mach Numbers," Experimental Data Base for Computer Program Assessment, AGARD-AR-138, pp. B1–1 B1–44, May 1979.
27. VATSA, V. N., SANETRIK, M. D., and PARLETTE, E. B., "Development of a Flexible and Efficient Multigrid-Based Multiblock Flow Solver," AIAA 93–0677, 1993.
28. ATKINS, H. L., "A Multiblock Multigrid Method for the Solution of the Euler and Navier-Stokes Equations for Three-Dimensional Flows," AIAA 91–0101, 1991.
29. ROSSOW, C. C., "Efficient Computation of Inviscid Flow Fields Around Complex Configurations Using a Multiblock Multigrid Method," Communications in Applied Numerical Methods, vol. 8, pp. 735–747, 1992.
30. THOMAS, J. L., WALTERS, R. W., REU, T., GHAFFARI, F., WESTON, R. P., and LUCKRING, J. M., "A Patched-Grid Algorithm for Complex Configurations Directed Toward the F/A-18 Aircraft," AIAA 89–0121, 1989.
31. CANNIZZARO, F. E., ELIMILIGUI, A., MELSON, N. D., and VON LAVANTE, E., "A Multiblock Multigrid Method for the Solution of the Three-Dimensional Euler Equations," AIAA 90–0105, 1990.
32. JAMESON, A., "Time Dependent Calculations Using Multigrid with Applications to Unsteady Flows Past Airfoils and Wings," AIAA 91–1596, July 1991.
33. JESPERSEN, D., "A Time-Accurate Multiple-Grid Algorithm," AIAA 85–1493, 1985.
34. ANDERSON, W. K., THOMAS, J. L., and RUMSEY, C. L., "Extension and Application of Flux-Vector Splitting to Calculations on Dynamic Meshes," AIAA Journal, vol. 27, June 1989.

35. MELSON, N. D., SANETRIK, M. D., and ATKINS, H. L., "Time-Accurate Navier-Stokes Calculations with Multigrid Acceleration," AIAA 93–3392, 1993.

36. SAAD, Y., and SCHULTZ, M., "GMRES: A Generalized Minimal Residual Algorithm for Solving Nonsymmetric Linear Systems," SIAM J. Sci. Statist. Comput., no. 7, pp. 856–869, 1986.

37. WIGTON, L., YU, N., and YOUNG, D., "GMRES Acceleration of Computational Fluid Dynamics Codes," AIAA 85–1494, 1985.

38. PULLIAM, T., "Euler and Thin-Layer Navier-Stokes Codes: ARC2D, ARC3D," Computational Fluid Dynamics Workshop at the University of Tennessee Space Institute, Mar. 1984.

39. CHOI, Y., and MERKLE, C., "Application of Time-Iterative Schemes to Incompressible Flows," AIAA Journal, vol. 23, no. 10, 1985.

40. TURKEL, E., "Preconditioned Methods for Solving The Incompressible and Low Speed Compressible Equations," ICASE Report No. 86–14, 1986.

41. TURKEL, E., "Review of Preconditioning Methods for Fluid Dynamics," ICASE Report No. 92–47, 1992.

42. VAN LEER, B., LEE, W., and ROE, P., "Characteristic Time-Stepping or Local Preconditioning of the Euler Equations," AIAA 91–1552–CP, June 1991.

43. GODFREY, A. G., *Topics on Spatially High-Order Accurate Methods and Preconditioning for the Navier-Stokes Equations with Finite-Rate Chemistry*, PhD thesis, Virginia Polytechnic Institute and State University, 1992.

44. ROE, P., "Approximate Riemann Solvers, Parameter Vectors, and Difference Schemes," J. of Comp. Phys., vol. 43, pp. 357–372, 1981.

45. JAMESON, A., SCHMIDT, W., and TURKEL, E., "Numerical Solutions of the Euler Equations by Finite Volume Methods Using Runge-Kutta Time-Stepping Schemes," AIAA 81–1259, 1981.

2.6 SPECIAL METHODS

This section is devoted to several variants in the solution of the Euler equations. These non conventional methods are:

1) Space-marching techniques for supersonic steady flow problems.

2) Shock-fitting techniques in which a shock wave is considered as a true discontinuity of the flow field represented either as an internal curve or surface in the computational mesh (floating shock tracking) or more often as a boundary of the computational domain (bow-shock fitting).

3) Inverse and design techniques which can be interpreted as the fitting of a vortex sheet (or a stream-surface in the steady case). This surface may be considered either as an interface between two flow fields both computed by solution of the Euler equations or, more generally, as the external boundary of one flow domain on which a pressure condition is prescribed (for a free-surface flow like an isobaric jet or for design purpose).

These methods are valuable alternatives to the use of the time-dependent, shock-capturing and direct (in the sense of fixed-boundary) solution of the Euler equations.

Indeed, the classical unsteady approach, described in the above subsections of this chapter, can be applied to special problems such as the steady-state solution of fully supersonic flows comprising or not internal and bow shocks or such as the design of a wall boundary to be iteratively modified by numerical optimization. However, in these special circonstances, it may be interesting to benefit from large savings in computing time (space-marching or inverse design methods) and from a noticeable increase in accuracy (shock-fitting techniques).

The discussion on the relative merits of standard and special methods is made more difficult by the uninterrupted progress of researchers finding new techniques which succeed in filling the gap between two opposite strategies. This is the case for example of the space-marching technique where the old method with the marching solution of the steady Euler equations (which in general requires a standard method for setting upstream conditions at a blunt nose) can be replaced efficiently by pseudo space-marching techniques using a modified unsteady Euler code (see below in subsection 2.6.1). This is also the case when combining shock-capturing for internal shocks and shock-fitting for the bow shock. Another example is given with the shock-fitting technique being eventually replaced by shock-capturing in association with mesh-fitting and adaption of either algebraic or variational type. Lastly, inverse techniques can be devised which make use of slightly modified direct solvers with a good efficiency (see below subsection 2.6.3).

In the following, we give some details on the special techniques mentioned above but it is remarkable that there are many common features and affinities between them and probably the more clear reason for that is their use of quasi-linear or characteristic forms of the Euler equations.

Therefore, after the presentation made in Chapter 1, we begin by a complement to the description of some characteristic forms of the Euler equations in order to describe the numerical treatment of the boundary conditions for shock-fitting and inverse design techniques.

Starting from the conservation law form:

$$\frac{\partial \mathbf{w}}{\partial t} + \frac{\partial \mathbf{F}}{\partial x} + \frac{\partial \mathbf{G}}{\partial y} + \frac{\partial \mathbf{H}}{\partial z} = 0 \qquad (2.6.1)$$

the quasilinear form (given in Eq.(1.4.2)) is:

$$\frac{\partial \mathbf{w}}{\partial t} + \mathbf{A}\ \frac{\partial \mathbf{w}}{\partial x} + \mathbf{B}\ \frac{\partial \mathbf{w}}{\partial y} + \mathbf{C}\ \frac{\partial \mathbf{w}}{\partial z} = 0 \qquad (2.6.2)$$

We define the matrix $\mathbf{A}_n$ as a linear combination of the Jacobian matrices $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$ formed with the components of $\vec{n}$, an arbitrary unit vector:

$$\mathbf{A}_n = n_x \mathbf{A} + n_y \mathbf{B} + n_z \mathbf{C} \qquad (2.6.3)$$

and to introduce a local orthonormal basis made of $\vec{n}$ and of two other unit vectors $\vec{\xi}$ and $\vec{\eta}$ so that a cartesian derivative is expressed (locally) in terms of derivatives in the new basis:

$$\frac{\partial}{\partial x_j} = n_j \frac{\partial}{\partial n} + \xi_j \frac{\partial}{\partial \xi} + \eta_j \frac{\partial}{\partial \eta} \qquad (2.6.4)$$

Due to the hyperbolicity of the Euler system of equations, the matrix $\mathbf{A}_n$ has real eigenvalues λ_i and a set of linearly independent left eigenvectors $\mathbf{m}_i$.

To get the required characteristic form of the equations (also called the compatibility relations), we form the linear combinations of Eq.(2.6.2) obtained by multiplication with each eigenvector:

$$\mathbf{m}_i^T \left(\frac{\partial \mathbf{w}}{\partial t} + \mathbf{A}_j \frac{\partial \mathbf{w}}{\partial x_j} \right) = 0 , \quad i = 1,\ldots 5 \qquad (2.6.5)$$

giving through the use of Eqs.(2.6.3-4):

$$\mathbf{m}_i^T \left(\frac{\partial \mathbf{w}}{\partial t} + \lambda_i \frac{\partial \mathbf{w}}{\partial n} \right) + \mathbf{m}_i^T \mathbf{R} = 0 \qquad (2.6.6)$$

where $\mathbf{R}$ involves only derivatives of $\mathbf{w}$ in the plane ξ , η.

Assuming that these derivatives are known, Eqs.(2.6.6) can be interpreted as transport equations for a plane wave moving with velocity $\lambda_i \vec{n}$.

The expression for the eigenvectors $\mathbf{m}_i$ is more complicated than for the eigenvectors $\mathbf{l}_i$ attached to the use of the primitive variables $\mathbf{q} = (\rho , \vec{v} , p)$, as described in section 1.4. With the slightly modified notations of Eqs.(2.6.6), we have here:

$$\mathbf{l}_i^T \left(\frac{\partial \mathbf{q}}{\partial t} + \lambda_i \frac{\partial \mathbf{q}}{\partial n} \right) + \mathbf{l}_i^T \mathbf{S} = 0 \qquad (2.6.7)$$

An important difference with Section 1.4 is that we have to consider a boundary moving with the velocity $\vec{u}_\Sigma$. The normal relative velocity of the fluid on Σ is:

$$v_r = (\vec{v} - \vec{u}_\Sigma) . \vec{n} = v_n - u_\Sigma \qquad (2.6.8)$$

The discussion for the number of numerical boundary conditions to use rests on the sign of $(\lambda_i - u_\Sigma)$ and on the value of the normal relative Mach number on Σ:

$$M_n^r = v_r / a \qquad (2.6.9)$$

Discretization of characteristic relations

Concerning the discretization of the characteristic relations given by Eqs.(2.6.6-7), it is convenient, like for the internal node discretization, to separate the spatial discretization from the time discretization. In fact, it is preferable to have a rather similar treatment for internal and boundary nodes, even though space derivatives in outgoing wave characteristic relations are necessarily one-sided and in general one order less accurate than for interior schemes.

A practical matching technique based on characteristic relations[1] which is of special interest for the fitting of discontinuities and quite easy to implement at least for explicit time stepping schemes, is described below for the case of cell-vertex or node-centered discretization schemes.

Two steps are considered. First a provisional value $\mathbf{w}^*$ is

computed on Σ at time $(n+1)$ from a complete discretized Euler system on Ω and Σ without taking into account any boundary condition. For simplicity, we restrict the presentation to a boundary treatment which is only of first order accuracy in time so that $\mathbf{w}^*$ can be written:

$$\mathbf{w}^* = \mathbf{w}^n - \Delta t \, (\delta \mathbf{F}_j / \delta x_j)^n \qquad (2.6.10)$$

where $\delta \mathbf{F}_j / \delta x_j$ is a discrete approximation of $\partial \mathbf{F}_j / \partial x_j$ on Σ (obtained using only the discrete values of $\mathbf{w}^n$ in Ω and on Σ).

The second step rests on the use of the characteristic relations of type (2.6.6) which are written in discretized form on Σ:

$$(\mathbf{m}_i{}^T)^n \, (\mathbf{w}^{n+1} - \mathbf{w}^*) = 0 \, , \quad i = 1,\ldots,5 \qquad (2.6.11)$$

where we have used Eq.(2.6.10).

However, due to the complicate expression of $\mathbf{m}_i$, it is convenient to replace these characteristic relations by those based on the non-conservative variables. After having deduced p^*, $\bar{v}^*$, ρ^* directly from $\mathbf{w}^*$, the system corresponding to Eq.(2.6.7) is the following:

$$\begin{aligned}
&a)\ \lambda_1 = v_n \ ; (p^{n+1} - p^* - (a^2)^n \, (\rho^{n+1} - \rho^*) = 0 \\
&b)\ \lambda_2 = v_n \ ; (v_\xi^{n+1} - v_\xi^*) = 0 \\
&c)\ \lambda_3 = v_n \ ; (v_\eta^{n+1} - v_\eta^*) = 0 \\
&d)\ \lambda_4 = v_n + a \ ; (p^{n+1} - p^*) + (\rho a)^n \, (v_n^{n+1} - v_n^*) = 0 \\
&e)\ \lambda_5 = v_n - a \ ; (p^{n+1} - p^*) - (\rho a)^n \, (v_n^{n+1} - v_n^*) = 0
\end{aligned} \qquad (2.6.12)$$

The characteristic relations for which $(\lambda_i - u_\Sigma) \geq 0$ have to be used whereas those correponding to $\lambda_i - u_\Sigma < 0$ should be replaced by physical boundary conditions from the outside of Ω.

These equations will be used below for shock-fitting and inverse methods.

2.6.1 Space-marching techniques

Steady supersonic flows have been studied numerically for a long time with a strong impetus given to the development of finite difference methods at the beginning of the 70's (after early studies based on the method of characteristics) by the first three-dimensional computations of flow fields past the Space Shuttle[2].

The governing Euler equations for the steady supersonic flows are hyperbolic with the flow direction as a time-like direction. Thus the numerical solution can be obtained by marching by plane in this direction (say the z-direction). For solving such a three-dimensional steady supersonic flow problem, it is natural to use a numerical method quite similar to those concerning the solution of a two-dimensional unsteady flow problem with time-dependent boundaries (either known or to be partly computed as a free-boundary). Therefore, a space-marching method can afford a considerable reduction in computer storage and in computer time in comparison with the unsteady approach in three dimensions.

Finite difference methods were first devised in non-conservative form with primitive variables ρ, $\bar{v}$, p [3] or with characteristic variables[4,5,6]:

$$P = ln(p) \, , \quad \bar{v} \, , \quad S = \gamma \, ln(T) - (\gamma - 1) \, P$$

For those methods, not only the bow shock but also internal shocks have to be fitted.

By contrast, other finite difference methods[2,7,8,9], mainly with capturing of internal shock waves, were developed based on the conservation law form of the Euler equations. They are generally written in cylindrical coordinates adequate for computation of flows past slender bodies which we choose here as a typical application of these techniques.

With u, v, w as components of $\bar{v}$ in z, r, θ coordinates, the equations write:

$$\frac{\partial \tilde{\mathbf{U}}}{\partial z} + \frac{\partial \tilde{\mathbf{P}}}{\partial r} + \frac{\partial \tilde{\mathbf{Q}}}{\partial \theta} + \tilde{\mathbf{R}} = 0 \qquad (2.6.13)$$

$$\begin{aligned}
\tilde{\mathbf{U}} &= (\rho u \, , \ p + \rho u^2 \, , \ \rho uv \, , \ \rho uw \,)^T \\
\tilde{\mathbf{P}} &= (\rho v \, , \ \rho uv \, , \ p + \rho v^2 \, , \ \rho vw \,)^T \\
\tilde{\mathbf{Q}} &= \frac{1}{r} \, (\rho w \, , \ \rho uw \, , \ \rho vw \, , \ p + \rho w^2 \,)^T \\
\tilde{\mathbf{R}} &= \frac{1}{r} \, (\rho v \, , \ \rho uv \, , \ \rho (v^2 - w^2) \, , \ 2 \rho vw \,)^T
\end{aligned} \qquad (2.6.14)$$

The steady character of the flow permits the replacement of the conservation law for total energy by the Bernoulli equation:

$$\frac{\gamma}{\gamma - 1} \, \frac{p}{\rho} + \frac{1}{2} \, (u^2 + v^2 + w^2) = H_\infty$$

The shock layer is bounded by the given body surface, $r = b(\theta, z)$, and by the bow shock wave $r = c(\theta, z)$, which is an unknown surface to be determined.

This flow domain (between two plane sections $z_0 \leq z \leq z_1$ can be mapped into a computational region $z_0 \leq Z \leq z_1$, $(X, Y) \in [0, 1]^2$ with:

$$\begin{aligned}
X &= X(z, r, \theta) \\
Y &= Y(z, r, \theta) \\
Z &= z
\end{aligned}$$

where X, Y are curvilinear grid coordinates stretched in each Z plane according to the flow features.

The governing equations Eqs.(2.6.13) when transformed in this computational space become:

$$\frac{\partial \mathbf{U}}{\partial Z} + \frac{\partial \mathbf{P}}{\partial X} + \frac{\partial \mathbf{Q}}{\partial Y} + \mathbf{R} = 0 \qquad (2.6.15)$$

with: $\mathbf{U} = \tilde{\mathbf{U}} / J$

$$\begin{aligned}
\mathbf{P} &= (X_z \tilde{\mathbf{U}} + X_r \tilde{\mathbf{P}} + X_\theta \tilde{\mathbf{Q}}) / J \\
\mathbf{Q} &= (Y_z \tilde{\mathbf{U}} + Y_r \tilde{\mathbf{P}} + Y_\theta \tilde{\mathbf{Q}}) / J \\
\mathbf{R} &= \tilde{\mathbf{R}} / J \ ; \quad J = X_z Y_\theta - X_\theta Y_r
\end{aligned}$$

The system of Eqs.(2.6.15) is hyperbolic in Z and it can be discretized in a same manner as an unsteady two-dimensional problem. A very common approach has been a discretization by finite differences and a solution with MacCormack explicit scheme[7,8,9,10]. Another more recent possibility is a discretization by finite volume techniques with a Godunov method[11] or with other upwind schemes. Solution with an explicit scheme proceeds by starting from a known solution at plane Z and computing a new solution in plane $Z + \Delta Z$ with boundary conditions[12,13] taking into account the evolution of the body shape and the change in the shock wave position by satisfying the Rankine-Hugoniot relations (see next Subsection).

One problem related to the use of the conservation law form is that the unknowns (conservative fluxes in the Z-direction) need to be "decoded" into the physical variables ρ, u, v, w, p in order to evaluate the transverse fluxes and

the source terms. For a perfect gas, this decoding reduces to a simple root finding problem. Starting from:

$$U_1 = \rho u \ , \ U_2 = p + \rho u^2 \ , \ U_3 = \rho uv \ , \ U_4 = \rho uw$$

we get:

$$p = U_2 - U_1 u \ , \ \frac{1}{\rho} = u/U_1 \ , \ v = U_3/U_1 \ , \ w = U_4/U_1$$

and by substitution in Bernoulli equation we obtain a second degree equation for u:

$$(\gamma+1)U_1^2 u^2 - 2\gamma U_2 U_1 u + (\gamma-1)(2U_1^2 H_\infty - U_2^2 - U_3^2) = 0$$

Only the largest root is meaningful, giving:

$$u = \frac{1}{(\gamma+1)U_1}(\gamma U_2 + \sqrt{\gamma^2 U_2^2 - (\gamma^2-1)(2U_1^2 H_\infty - U_2^2 - U_3^2)})$$

Another question related to the space-marching technique is the step size limitation in the Z-direction associated with an explicit scheme.

A CFL condition links ΔZ to the space step size in the cross-flow plane, typically:

$$\Delta Z \ \leq \ \Delta X \ / \ \sigma(M)$$

with σ is the maximum of local eigenvalues of a matrix M of the form $C^{-1}A$ with the notations of Eq.(2.6.2). This limitation in the step size for marching in space is not too severe in zones where the body section has fast variations but it can be considered as too costly when the flow is very smooth in the Z-direction like in the case of nearly conical flows.

The use of implicit stepping for space-marching schemes is not in general use for solving the steady Euler equations when the bow shock is fitted. Indeed, there is a limited interest in using an implicit scheme for interior points if an explicit treatment of the bow shock positioning yet restricts the maximum step size in the marching direction.

Fully implicit treatment of both the interior point calculation and the bow-shock adjustment was shown to be very efficient on a one-dimensional flow in a variable area duct with an internal shock wave which was computed for a steady solution via a time asymptotic approach[14]. Such a method could clearly be used for a supersonic conical axisymmetric flow. Extension to one more space dimension should be directly applicable to the solution of conical flows for example past conical wings at angle of attack, but should not be so much necessary for non conical bodies where accuracy in the Z direction cannot be obtained with too large values of ΔZ.

If the bow shock is captured and the wall boundary condition is treated implicitly, an implicit scheme for space marching is very efficient and it appears even more useful for the solution of Parabolized Navier-Stokes (PNS) equations due to the severe CFL restriction on the ΔZ with an explicit scheme and the very small grid spacing needed at the wall.

In the case of a blunt body, a supersonic space-marching code must be completed by an other one capable of computing the subsonic flow region between the detached shock wave and the body nose, thus providing the necessary initial solution of the space-marching procedure.

Moreover, realistic high speed flight vehicle configurations often give rise to subsonic pockets inside the shock layer. The conventional space-marching method then fails for such flows and it can be necessary to combine a space-marching technique for supersonic parts and an unsteady Euler solver for subsonic parts with delicate problems of different grid systems to be coupled.

Space-marching procedures by local iteration

To avoid this coupling, several authors have proposed a "unified approach" based on unsteady Euler equations with a plane by plane strategy and local or global iterations according to the fully supersonic or mixed character of the flow.

Although some investigations about this idea were led in the 70's with the conventional MacCormack scheme, the full development of this technique found its actual start in the middle of the 80's[15-19] in relation with upwind discretization and relaxation methods for implicit time-stepping[20,21].

These now well-established techniques have revealed to be both efficient for obtaining numerical solutions over realistic configurations and very easy to implement in a 3-D code. They provide the possibility of building the three-dimensional mesh system before the flow calculation, which can be of interest for a better control of the grid stretching or adaption. Besides that, the further development which consists in taking into account a discretization of viscous terms to be added to the Euler code is rather straightforward; this development leads to a PNS-like approximation. We will focus our attention hereafter on these "pseudo space-marching" techniques based on the unsteady Euler equations.

Space-marching techniques making use of the unsteady Euler equations have mainly been developed using the finite-volume TVD discretization. They are based on either explicit or implicit time differencing, and suppose a "plane by plane" organization of the computation.

Explicit Approach

The simplest approach is based on an explicit upwind scheme as described in Ref.19. The space-marching procedure simply amounts to run out, until convergence, the 3-D time-dependent explicit scheme plane after plane, with the upstream conservative variables in the plane k-1 fixed at their previously computed value, and with the downstream values in the plane k+1 extrapolated from the upstream and current values.

Basically, in the notations of Eq.(2.6.2) with cartesian coordinates, the explicit algorithm can be written as:

$$\frac{\mathbf{w}_{i,j,k}^{l+1} - \mathbf{w}_{i,j,k}^{l}}{\Delta t_k} + (\mathbf{F}_{i+1/2} - \mathbf{F}_{i-1/2})^l + (\mathbf{G}_{j+1/2} - \mathbf{G}_{j-1/2})^l$$

$$+ \ (\mathbf{H}_{k+1/2})^l - (\mathbf{H}_{k-1/2})^l = 0 \ ; \quad l = 0, 1, \ldots l_{max} \qquad (2.6.16)$$

where $\mathbf{w}$ is the solution vector of the conserved quantities, $\mathbf{F}$, $\mathbf{G}$, $\mathbf{H}$ are the numerical fluxes at the sides of the control volume (i,j,k) including metric terms, and k is the marching direction. Clearly, Eqs.(2.6.16) represent an iterative process which, if converged, provides a discretized solution of the steady Euler equations. Since the $\mathbf{F}$ and $\mathbf{G}$ flux evaluation depends explicitely on $\mathbf{w}_k^l$, the only delicate point is the definition of $(\mathbf{H}_{k+1/2})^l$. In order to get conservation, the flux $(\mathbf{H}_{k-1/2})^l$ has to be computed with the same formula as the one used for $(\mathbf{H}_{k-1/2})^*$ at the previous $k-1$ plane calculation (a $'*'$ denotes here a frozen upstream quantity).

We assume that a flux vector splitting (FVS) scheme is used for the numerical flux evaluation:

$$(\mathbf{H}_{k+1/2})^l = \mathbf{H}^+(\mathbf{w}_{k+1/2}^L)^l + \mathbf{H}^-(\mathbf{w}_{k+1/2}^R)^l \qquad (2.6.17)$$

with $\mathbf{w}^L$ and $\mathbf{w}^R$ the left and right values for the $i,j,k+1/2$ cell face. Actually, the second term remains at zero for a fully supersonic flow. Therefore, we can get:

a) a first order scheme with

$$(\mathbf{w}_{k+1/2}^L)^l = \mathbf{w}_{i,j,k}^l$$

b) a second order scheme with

$$(\mathbf{w}^{L}_{k+1/2})^{l} = \mathbf{w}^{l}_{i,j,k} + \frac{\Delta Z}{2}(\mathbf{w}^{l}_{i,j,k} - \mathbf{w}^{*}_{i,j,k-1})$$

Initialization for $\mathbf{w}^{0}_{i,j,k}$ follows easily either simply with $\mathbf{w}^{0}_{k} = \mathbf{w}^{*}_{k-1}$ or from a higher order extrapolation.

In contrast with a pure space-marching scheme, the stability of the process depends not only on ΔZ_k but also from Δt_k which remains limited by a CFL condition for a small ΔX_k but permits an increased ΔZ_k.

This local iteration with an explicit time-stepping permits significant gains in efficiency with respect to a global iteration and improvements in accuracy, robustness and programming simplicity with respect to a classical pure space-marching scheme.

Implicit Approach

This approach[15,16,17,18] rests on a planar Gauss-Seidel relaxation scheme combined with an approximate factorization in the plane Z_k. It can be derived from:

$$\frac{\mathbf{w}^{l+1}_{ijk}-\mathbf{w}^{l}_{ijk}}{\Delta t_k} + (\mathbf{F}_{i+1/2} - \mathbf{F}_{i-1/2})^{l+1} + (\mathbf{G}_{j+1/2} - \mathbf{G}_{j-1/2})^{l+1} + (\mathbf{H}_{k+1/2} - \mathbf{H}_{k-1/2})^{l+1} = 0 \qquad (2.6.18)$$

Using a "delta" formulation, an implicitation of only the first order terms and a linearization of the Jacobians, Eq.(2.6.18) leads to the following matrix system to be solved:

$$\begin{cases} 1/\Delta t_k\ \Delta \mathbf{w}_{ijk} + A_i \Delta \mathbf{w}_{i-1,j,k} + D_i \Delta \mathbf{w}_{ijk} + B_i \Delta \mathbf{w}_{i+1,j,k} \\ \quad + A_j \Delta \mathbf{w}_{i,j-1,k} + D_j \Delta \mathbf{w}_{ijk} + B_j \Delta \mathbf{w}_{i,j+1,k} \\ + A_k \Delta \mathbf{w}_{i,j,k-1} + D_k \Delta \mathbf{w}_{ijk} + B_k \Delta \mathbf{w}_{i,j,k+1} = RHS_{ijk} \end{cases} \qquad (2.6.19)$$

where RHS represents the explicit part of the scheme and

$$\Delta \mathbf{w} = \mathbf{w}^{l+1} - \mathbf{w}^{l}.$$

The pseudo space-marching approximation leads to:

$$\Delta \mathbf{w}_{i,j,k-1} = 0 \quad \text{and} \quad \Delta \mathbf{w}_{i,j,k+1} = \Delta \mathbf{w}_{i,j,k},$$

which amounts, in Eq.(2.6.19), to cancel the term A_k and to add the term B_k to the diagonal D_k. In order to get a symmetric formulation in the case of subsonic pockets, the term B_k is cancelled according to Ref.15.

The overall implicit approximate factorization procedure can be summed up, in matrix formulation, as:

$$(I + \tilde{D}_k^{-1} M_i)\ (I + \tilde{D}_k^{-1} M_j)\ \Delta \mathbf{w} = \tilde{D}_k^{-1}\ RHS \qquad (2.6.20)$$

where M_i, M_j represent the matrix terms along the crossflow directions, and

$$\tilde{D}_k = I/\Delta t_k + D_k$$

Realistic configurations require the use of multi-domain techniques. For relatively simple configurations the multi-domain gridding can be restricted to the marching (supersonic) direction[19]. The general multi-domain gridding associated with the implicit approach opens the door to the computation of complex supersonic flows such as a vortical structure at the leeward of an hypersonic delta wing[22] or vehicle configurations such as a realistic fighter configuration or the Space Shuttle Orbiter[15].

2.6.2 Shock-fitting techniques

Shock-fitting techniques are founded on the choice of representing a shock wave as a boundary of the computational domain or eventually as an interface between two such domains. They are used when the presence and the general shape of a shock is a priori known as a key feature of the flow solution. This can be the case in a divergent choked nozzle and even in a transonic or supersonic external flow past an airfoil, however the most frequent use of shock-fitting corresponds to the fitting of a bow-shock in front of the nose of a vehicle or of a projectile in supersonic flight. Another interest of such techniques, when no other shock is present inside the computational domain, is the possibility of choosing a convenient non-conservative set of equations (in place of the conservative formulation needed for shock-capturing). However, this possibility has been mainly used in two dimensions.

We describe here a general methodology to treat the fitting of such a shock wave surface Σ.

Let Ω_1 be the upstream domain, with unit normal $\bar{n}_1$ pointing downstream with $\bar{u}_\Sigma \cdot \bar{n}_1$ the normal velocity of the shock. For Ω_1, Σ is a supersonic outflow boundary ($M^r_{n_1} > 1$) whereas for Ω (the downstream domain, with normal $\bar{n}$ pointing outwards of Ω), it is a subsonic inflow boundary ($-1 < M^r_n < 0$). Therefore, on the upstream side of Σ, all the flow quantities either are computed from the full set of the discretized Euler equations $q_1^{n+1} = q_1^{*}$ or they are given by the freestream supersonic flow conditions in the general case of bow-shock-fitting.

On the downstream side of Σ, only Eq.(2.6.12-d) must be used:

$$(p - p^{*}) + (\rho a)^{n} (v_n - v_n^{*}) = 0 \qquad (2.6.21)$$

omitting the superscript $(n+1)$ on p and v_n.

The supplementary conditions are the Rankine-Hugoniot jump relations (Eq.(1.3.5)) with $u_\Sigma = \bar{u}_\Sigma \cdot \bar{n}$:

$$\begin{aligned} &a) \quad \rho (v_n - u_\Sigma) = \rho_1 (v_{1n} - u_\Sigma) = Q \\ &b) \quad p + Q\ v_n = p_1 + Q\ v_{1n} \\ &c) \quad \bar{v} \cdot \bar{\xi} = \bar{v}_1 \cdot \bar{\xi} \\ &d) \quad \bar{v} \cdot \bar{\eta} = \bar{v}_1 \cdot \bar{\eta} \\ &e) \quad p\ v_n + Q\ E = p_1\ v_{1n} + Q\ E_1 \end{aligned} \qquad (2.6.22)$$

where the superscript $(n+1)$ has been omitted on all variables.

These jump relations together with Eq.(2.6.21) appear as a system of 6 equations for the 6 unknowns ρ, $\bar{v}$, p, u_Σ. This non-linear system is solved by an iterative method based on successive approximations of u_Σ (thus of Q) starting with the known value u_Σ^n. For a given Q, the values of p and v_n are determined from Eq.(2,6.21) and Eq.(2.6.22 b). Tangential velocity components are given by Eq.(2.6.22 c,d). Then Eq.(2.6.22 e) is solved for ρ (after expressing E in terms of the non-conservative variables) and a new value u_Σ is determined from Eq.(2.6.22 a) leading to a new iterative step.

In the case of fitting a bow-shock upstream of a body, the computational domain Ω can be meshed with a body-fitted grid moving like a concertina between the bow shock and the body with all nodes sliding along fixed lines roughly normal to the body and the shock. This family of normal lines has to be generated in a preprocessing step by any method giving a regular body-conforming mesh system between the vehicle and an outer surface which must be far enough from the body to be upstream of the bow shock. The normal family of grid lines is replaced by smooth curves obtained by piecewise polynomial interpolation with an explicit parametrization.

The trace of the shock on each of these curves provides a definition of the boundary moving at each time step at a velocity calculated according to the method indicated above.

A straightforward implementation of the interlaced grid motion and solution evolution is given below.

For a steady flow, calculation of the solution until convergence consists, at each time step Δt, in successively executing the three following stages:

i) The flow solution at time t^{n+1} is obtained from values at time t^n by applying the basic numerical scheme in a fixed mesh M^n which coincides with the moving mesh at time t^n.

ii) The new boundary is determined by using the normal velocity of the shock calculated as indicated above. Then interior nodes are defined giving the mesh M^{n+1} at time t^{n+1}.

iii) The flow quantities at time t^{n+1} on the mesh M^n are projected onto the mesh M^{n+1} using a first order Taylor expansion:

$$\phi(t^{n+1}, M^{n+1}) = \phi(t^n, M^n) + \Delta t\ \nabla\phi(t^{n+1}, M^n).\vec{u} \qquad (2.6.23)$$

where ϕ is any flow variable and where $\vec{u}$ denotes the mesh point velocity which is tangent to the given mesh lines.

Due to the independence between the stages ii)-iii) and the stage i), this process is easy to implement in an existing code written with an explicit scheme for a fixed grid.

Other techniques have been used for the bow-shock fitting problem and in particular the coordinate transformation method which consists in using the mapping of the moving physical domain to the fixed computational domain and working directly on the transformed equations in these fixed coordinates and on the non-linear boundary conditions on the image of the shock boundary to iterate towards a steady solution.

A detailed discussion of the relative merits for different possibilities of mixing the use of an explicit or an implicit scheme for interior nodes with the weak or strong coupling between flow solution and shock tracking can be found in Ref.14.

2.6.3 Inverse design and free-surface flows

Free surface flows, shock-fitting methods, inverse and design problems all belong to the same mathematical class of free-boundary problems as opposed to the more usual fixed-boundary problems.

For all those flow problems, a part of the boundary limiting the domain occupied by the fluid is a priori unknown and has to be found during the solution process.

By comparison with a fixed-boundary problem, it is clear that the relaxation of the parameters defining the position of the flow domain boundary yields a larger class of solutions among which the selection of a particular one may result either from the optimization of some criterion or from the prescription of a supplementary boundary data (in principle the pressure) in order to get uniqueness.

The first case corresponds to optimum design problems, whereas the second is usually named an inverse problem. A detailed classification of the various methods for solving these problems can be found in Ref. 23 whereas several of them are described in Refs. 24 and 25.

Optimum design problems are generally solved by coupling a "direct solver" (a solver with a given fixed boundary), a boundary shape and grid updating algorithm and a numerical optimization code allowing to progressively modify the boundary shape until an optimum is reached. The definition of the criterion to be minimized opens many possibilities based either on physical or mathematical principles. In numerical optimization techniques for optimum design the direct solver is most often used as a "black box" and the qualities that a good design requires for such a solver are: short time response, low sensitivity to its numerical discretization parameters and high sensitivity to the aerodynamic design parameters.

Therefore numerical optimization is not yet frequently used with Euler direct solvers. Indeed, for the design of a transonic transport wing, it is often said that full-potential equations with a simple quasi three-dimensional integral boundary layer correction is adequate for solving this problem [26]. However, for the case of purely supersonic flows, Euler solvers can be used with benefit has shown in Section 4.8 where design applications by Schonc[27] and by Rieger[28] are presented. It is quite clear that with the progress of Euler codes as analysis tools, their usefulness will be increasing for design problems treated by numerical optimization.

The second category of free-boundary problems results from the specification of a flow quantity on the unknown boundary. We shall concentrate the present discussion on this kind of problem and its discretization.

We consider the problem of a fluid flow in a domain Ω with a boundary Γ which comprises a slip-line or slip-surface Σ, not a priori known and which must be found as a part of the solution process. This boundary surface is considered as a material surface. In other words, if a fluid particle is on the free surface it forever remains on it. Let assume that Σ can be represented by a single-value function:

$$z = \beta(x, y, t) \qquad (2.6.24)$$

Then the above assumptions mean that there is a kinematical boundary condition holding on Σ:

$$\frac{d\beta}{dt} = \frac{\partial\beta}{\partial t} + u\frac{\partial\beta}{\partial x} + v\frac{\partial\beta}{\partial y} = w_\Sigma \qquad (2.6.25)$$

The other condition is a pressure condition:

$$p = \pi(x, y, t) \qquad on\ \ \Sigma \qquad (2.6.26)$$

The simplest case is $\pi = constant$ for a free-surface flow like in the hydrodynamic ship wave problem studied in Ref. 29 or for an engine jet with external flow at rest [30,31].

The case $\pi = f(x, y)$ can correspond to an inverse design problem where a pressure distribution is prescribed on a part Σ of a body limiting the flow domain Ω[32-36].

As in the previous Subsection on shock-fitting, there are two main possibilities for treating the moving mesh problem with curvilinear body-fitted grids:

1) Working in the physical moving domain Ω with three stages at each time step towards a steady state solution[31].

2) Working in the computational domain after a transformation of coordinates from x, y, z to ξ, η, ζ and solving the transformed flow equations and free-surface conditions in alternate stages at each time step [29].

It is also possible to replace the alternate stages of solving the flow equations with a given provisional boundary shape and then of updating the position of the boundary, by a simultaneous solution in a strong coupling between these two stages but at the cost of a more complex solution method. This is the case for the Newton solution of the mixed inverse method of Drela and Giles [32].

The most important points are in any case the choice of the boundary conditions and of the updating method for the position of the material surface Σ.

Coming back to Eq.(2.6.12), we are in the situation where the relative fluid velocity is zero ($v_n = u_\Sigma$) and only the characteristic relations corresponding to $\lambda_5 = v_n - a$ has to be replaced by the pressure boundary condition Eq.(2.6.26), independently of the subsonic or supersonic character of the flow in the tangential direction in Ω along Σ.

At each time step, the flow solution can be solved with these boundary conditions on Σ, then from the relation b),c) and d) in Eq.(2.6.12) and from the known value of p^{n+1}, the fluid velocity is obtained at time $n+1$. Thus a new value of $\vec{u}_\Sigma$ is available which can be used with Eq.(2.6.25) to get a new position of Σ at time $n+1$.

This system of equations for the unknown free surface is discretized in a straightforward manner but must include a dissipation term at least in the flow direction (or an upwinding differencing) in order to avoid decoupling of the solution with purely centered differencing [29,31].

2.6.4 References

1. VEUILLOT, J.P. and CAMBIER, L. "A Sub-Domain Approach for the Computaion of Compressible Inviscid Flows", *Proceedings of the INRIA Workshop on "Numerical Methods for the Euler Equations of Fluid Dynamics"*, SIAM Philadelphia, 1985.

2. KUTLER, P., LOMAX, H. and WARMING, R.F., "Computation of Space Shuttle Flow Fields Using Noncentered Finite Difference Schemes", AIAA-Paper 72-193, Jan. 1972.

3. THOMAS, P.D., VINOKUR, M., BASTIANON, R.A. and CONTI, R.J., "Numerical Solution for Three-Dimensional Inviscid Supersonic Flow", AIAA Journal, Vol. 10, No.7, pp. 887-894, July 1972.

4. MORETTI, G., GROSSMAN, B. and MARCONI F., "A Complete Numerical Technique for the Calculation of Three Dimensional Inviscid Supersonic Flows", AIAA Paper 72-192, 1972.

5. MARCONI, F. and SALAS, M., "Computation of Three-Dimensional Flows about Aircraft Configurations", Computers and Fluids, Vol.1., pp. 185-195, 1973.

6. SICLARI, M.J., "Investigation of Cross-Flow Shocks on Delta Wings in Supersonic Flow', AIAA Journal, Vol.18, No.1, Jan. 80.

7. KUTLER, P., REINHARDT, W.A. and WARMING, R.F., "Multishocked Three Dimensional Supersonic Flowfields with Real Gas Effects", AIAA Journal Vol. 11, NO.5, pp. 657-664, May 1973.

8. SOLOMON, J.M., CIMENT, M., FERGUSON, R.E. and BELL, J. B., "Inviscid Flowfield Calculation for Re-entry Vehicles with Control Surfaces", AIAA Journal, Vol. 15, No.12, pp.1742-1749, Dec. 1977.

9. WARDLOW, A.B.Jr, SOLOMON, J.M. and BALTAKIS, F.P.,"Supersonic Inviscid Flowfield Computations of Missile Type Bodies", AIAA Journal, Vol.19, No.7, July 1981.

10. KUTLER, P. "Computation of Three-Dimensional Inviscid Supersonic Flows", in *"Progress in Numerical Fluid Dynamics"*, Lecture Notes in Physics, Vol. 41, Springer Verlag, 1975.

11. WARDLAW, A.B.Jr and DAVIS, S.F., "A Second Order Godunov Method for Tactical Missiles", AGARD CP-412, Paper 12, April 1986.

12. KENTZER, C.P., "Discretization of Boundary Conditions on Moving Discontinuities", *Proceedings of the 2nd International Conference on Numerical Methods in Fluid Dynamics, Lecture Notes in Physics, Vol.8*, Springer Verlag, pp. 108-113, 1971.

13. ABBETT, M.J., "Boundary Condition Calculation Procedures for Inviscid Supersonic Flowfields", *Proc. AIAA CFD Conference, Palm Springs, Ca*, pp. 153-172, 1973.

14. BELL, J.B., SHUBIN, G.R. and SOLOMON, J.M., "Fully Implicit Shock Tracking", Jounal of Computational Physics, 48, pp. 223-245, 1982

15. CHAKRAVARTHY, S.R. and SZEMA, K.Y. "An Euler Solver for Three-Dimensional Supersonic Flows with Subsonic Pockets," AIAA Paper 85-1703, Cincinnati, July 1985.

16. WALTERS, R.W. and DWOYER, D.L., "An Efficient Strategy Based on Upwind/Relaxation Schemes for the Euler Equations", AIAA Paper 85-1529, July 1985.

17. THOMAS, J.L., van LEER, B. and WALTERS, R.W., "Implicit Flux-Split Schemes for the Euler Equations", AIAA Paper 85-1680, July 1985.

18. WALTERS, R.W. and DWOYER, D.L., "Efficient Solutions to the Euler Equations for Supersonic Flow with Embedded Subsonic Regions", NASA Tech. Paper 2523, Jan. 1987.

19. BORREL, M., MONTAGNE, J.L., DIET, J., GUILLEN, Ph. and LORDON, J. "Upwind Scheme for Supersonic Flows around Tactical Missiles," La Recherche Aerospatiale, English edition 1988-2.

20. CHAKRAVARTHY, S.R. "Relaxation Methods for Unfactored Implicit Upwind Schemes", AIAA Paper 84-0165, Jan. 1984.

21. VAN LEER, B. and MULDER, W.A., "Relaxation Methods for Hyperbolic Equations", Rep. 8420, Dep. Math. & Inform. Delft Univ. of Technol. 1984.

22. GUILLEN, Ph. and BORREL, M. "Hypersonic Delta Wing flow Calculations Using a Multidomain MUSCL Euler Solver," *Proceedings of Workshop on Hypersonic Flows for Re-entry Problems*. Desideri, J.A., Glowinski, R. and Periaux, J. Eds. Springer Verlag, Vol 1. 1991.

23. DULIKRAVICH, G.S., "Aerodynamic Shape Design and Optimization, Status and Trends", Journal of Aircraft, Vol. 29, No.6, Nov.-Dec. 1992.

24. *Proceedings of AGARD Specialist's Meeting on Computational Methods for Aerodynamic Design (Inverse) and Optimization.*", Loen, Norway, May 1989, Slooff J.W. Ed.,AGARD CP-463, 1990.

25. *Proceedings of a Special Course on Inverse Methods for airfoil Design for Aeronautical and Turbomachinary Applications*, Van Dem Braembusche R. Ed., AGARD Rept 780, 1990.

26. HICKS, R.H., CLIFF, S.E., MELTON, J.E., LANGLIN, R.G., GOODSELL, A.M., ROBERTSON, D.D., MOYER, S.A., "Euler and Potential Computational Results for Selected Aerodynamics Configurations" in *"Applied Computational Aerodynamics"*, P.A.Henne, Ed., Progress in Astronautics and Aeronautics, Vol 125, 1990.

27. SCHONE, J.,"Design of Supersonic Wing Using an Optimization Strategy Coupled with a Solution Scheme for the Euler Equations.", AIAA Paper 90-3060, 1990.

28. RIEGER, H., "Aerodynamics of Hypersonic Lifting Vehicles", AGARD CP-428, 1988.

29. FARMER, J., MARTINELLI, L. and JAMESON, A., "A Fast Multigrid Method for Solving Incompressible Hydrodynamic Problems with Free Surfaces", AIAA Paper-93-0767, Reno, NV, Jan. 1993.

30. VIVIAND, H. and VEUILLOT, J.P., "Methodes pseudo-instationnaires pour le calcul d'ecoulements transsoniques", Publicaton ONERA No 1978-4, English Translation, ESA TT 549.

31. COUAILLIER V. and VEUILLOT, J.P., "Inverse Methods for the Euler Equations", *Proceedings of the 2nd International Conference on Inverse Design Concepts and Optimization in Engineering Sciences, ICIDES*, G.S.Dulikravitch Ed., Dept. of Aerospace Engineering Penn State Univ., University Park, PA, pp131-146, Oct. 1987.

32. GILES, M. B. and DRELA, M., "Two-Dimensional Transonic Aerodynamic Design Method", AIAA Journal, Vol.25, No9, Sept. 1987.

33. MEAUZE, G., "Methode de calcul aerodynamique inverse pseudo-instationnaire", La Recherche Aerospatiale, 1980-1, English Translation, ESA TT 651, pp 23-30, 1980.

34. MEAUZE, G. "An Inverse Time Marching Method for the Definition of Cascade Geometry", Journal of Engineering Power, Vol. 104, pp. 650,656, July 1982.

35. ZANNETTI, L., "Time Dependent Method to Solve Inverse Problems for Unternal Flows", AIAA Journal, Vol. 18, July 1980.

36. ZANNETTI, L. and LAROCCA, F., "Inverse Methods for 3D Internal Flows, *AGARD Rept No 780*, Paper 8, 1990.

Chapter 3

Survey of Major Individual Euler Codes and their Capabilities

The increasing importance of computational aerodynamic methods, based on the Euler equations, in industrial design work is evident from the widespread development activities in all major aerospace industries.

In the following sections an attempt is made to gather all publicly available information on relevant Euler codes currently in industrial use in the NATO countries. Other countries, with the exception of Sweden and Switzerland, have not been considered and were not in the scope of this publication. Although a great deal of time was invested in accumulating this information we realize that other codes may exist in industrial sectors and research establishments that we are not aware of. Nevertheless, we believe that most of the codes are represented, which enables us to call this collection an Euler-code compendium.

In addition to a short overview of the capabilities of the individual codes, the essential characterisitics are also compiled in tables. Related references are pointed out and a person serving as point of contact (POC) for questions is listed for each code. We have tried to keep the list as concise as possible. The presentation is focused on general purpose codes with respect to their capability in treating 3-D complex problems in a regime that ranges from subsonic to supersonic flow. Based on recent projects in the United States and Europe, many codes also have the capability to handle hypersonic flow problems and can address gas states in chemical and thermodynamical equilibrium and in non equilibrium.

The following codes focus mainly on external aerodynamics, but some of them are also applicable for internal flow, especially on turbomachinery flow problems. Although the basic theory for treating numerically inviscid flow problems is outlined in previous chapters it has to be pointed out that a thorough treatment of the specific aspects of turbomachinery flow simulation can be found in the publication of another AGARD working group under the editorial supervision of Prof. Ch. Hirsch, VUB-Brussels.

A high percentage of codes are based on structured multiblock grid approaches that rely on a conservative finite-volume discretization technique, a numerical approximation scheme being outlined in the late 70's. The success of the methodology in the aerospace industry can be attributed to the fact that a discretization and time integration scheme was made available early which proved to be robust, simple and accurate enough to be accepted in industrial environments. Therefore, the landmark paper of Jameson, Schmidt and Turkel[1] stood as initial baseline for many first efforts in the field of inviscid flow simulation by solution of the Euler equations. In the meantime the concepts for numerically approximating the convective terms in the governing equations have been improved and refined and have been implemented in many advanced general-purpose codes.

Essential to the success of numerical field methods for solution of the steady-state inviscid flow equations is the invention of the so-called multigrid technique as one of the most effective and cost-efficient convergence acceleration methods known today. The improvements in performance of computer hardware over the last 20 years together with such algorithmic quantum steps, has enabled the introduction of Euler methods into routine industrial design and optimization cycles.

However, as the strength of the codes continue to increase with respect to their ability to treat geometrically complex flow problems, structured multiblock mesh generation appears to be the costly bottleneck in the Euler flow-simulation business.

In the face of high investments in sufficient cost-efficient grid generation systems, strong efforts in the development of unstructured-grid approaches have been supported to circumvent a possible deadlock between industrial task requirements and costs. An early view of emerging problems have led to the systematic development of finite-element methods for fluid flow problems at INRIA in France. However, it seems now that unstructured approaches, based on finite-volume schemes, are likely to achieve a similar success and acceptance in industry as the structured multiblock approaches have both now and in the past. In this context it may be of some interest that past and future trends in numerical techniques are discussed in recent overviews.[2,3]

The inviscid methods in industrial use today have reached a sufficient maturity and can be applied with confidence. Therefore, in Chapter 4 we have provided a limited but representative selection of simulation results that spans a fairly wide field of practical applications.

Highly accurate 2-D datum solutions past airfoils (section 4.1) and 3-D computational results of flows around wings (section 4.2) are presented as well as the capability of the Euler equations to capture vortical flows (section 4.3). The challenging aim of the 80's to treat complete air-vehicle configurations (section 4.4) has been achieved by some of the codes described subsequently. Euler simulations address also many relevant issues to propeller and propulsion flow problems as outlined in section 4.5 and are of importance for design support of space transportation systems that operate at hypersonic speeds (section 4.6). Applications for 3-D unsteady flows are emerging (section 4.7) as well as early attempts at integrating Euler solvers into optimization procedures to improve the design for relevant design parameters (section. 4.8).

It has to be pointed out that all results presented have been gained by the methods and codes described subsequently. Despite the fact that the results obtained so far are impressive, industrial needs and pressure to reduce costs will push developments of those methodologies that no longer require the permanent interaction of CFD experts and, therefore, can be operated directly by the design team. Evidence that support this conclusion can already be seen in the code compilation.

1. JAMESON A., SCHMIDT, W. and TURKEL, E. "Numerical Solutions for the Euler Equations by Finite Volume Methods Using Runge-Kutta Time Stepping Schemes", AIAA Paper 81-1259, June 1981.

2. *"CFD - Part I , An Assessment of Critical Technologies,"* *Aerospace America*, pp. 16-61, January 1992.

3. JAMESON A. "Numerical Wind Tunnel - Vision or Reality," AIAA-Paper 93-3201, 1993.

3.1 CODES FROM NORTH AMERICA

3.1.1 Overview

This section tabulates computer codes that are being used to solve the Euler equations in the United States and Canada. The codes are developed and are available through university, government, and industrial organizations. The majority of the algorithm developments have occurred at government research laboratories and through government-sponsored university research; the resulting codes, often general-purpose algorithms, are in the public domain and are widely disseminated. The codes developed by industry are often tailored to a specific need and are usually proprietary software. A general discussion of the distinguishing capabilities of the codes are discussed below.

Discretization Technique
In their basic discretization, the majority of the codes use the finite-volume technique or the closely related Galerkin finite-element method; the rest are finite-difference discretizations. Approximately half of the codes solve for the flow variables as cell-centered quantities, as opposed to cell-vertex or node-centered variables. For codes designed for unsteady flow simulations, the cell-centered finite-volume approach is most commonly used.

Grids
In terms of grids, the recent trend has been toward the development of unstructured grid methods; for 2-D Euler applications, these methods are sufficiently developed that to be the method of choice, largely because of the generality and simplicity of the grid generation. For 3-D Euler applications, the unstructured-grid codes are demonstrated to be competitive with the structured-grid methods. The adaptive-grid method is incorporated in several of these codes (NS72, FUN2D, LaRC-1, WL/Leggo) to increase the efficiency of the unstructured-grid codes considerably. The adaptive-grid codes generally use tetrahedral cells in three-dimensions, although the WL/Leggo code adapts through the continual refinement of hexahedral cells. The LaRC-1 code incorporates both temporal and spatial adaption capabilities for the time-accurate simulation of 3-D flows.

The structured-grid methods have been under development and in use longer than the unstructured-grid methods for aerodynamic applications and are highly evolved. However, the generation of a multiblock structured-grid generally requires considerably more man hours than the computer time required to generate the solution to either the Euler or Navier-Stokes equations on the resulting grid. The majority of the codes listed are, in fact, multiblock structured-grid codes. The multiblock structured-grid codes are also those that form the basis for most general-purpose codes for solving the Euler or Navier-Stokes equations. The patched grid method is used in several general-purpose codes (CFL3DE, F3D, UTRC-1) to simplify the task of grid generation for complex configurations and is used in several of the space-marching solvers (EMTAC-MZ, CFL3DE). The overset-grid method is less frequently available and is included in only two of the codes listed. The monoblock structured-grid codes are generally special-purpose codes, such as the shock-fitting GAUSS2D code or the finite-rate chemistry SPARK3D code.

Space/Time Discretization
The majority of the codes use the method-of-lines technique to decouple the spatial discretization from the temporal discretization. The advantage to this approach is that the steady state is independent of the time step. The coupled space/time discretization methods are predictor-corrector MacCormack schemes. The SPARK3D code is a predictor-corrector scheme with second-order-accurate temporal differencing and second- or fourth-order-accurate spatial accuracy. It is the only code listed with a spatial accuracy greater than second order. The temporal accuracy of the codes listed is no greater than second order.

The spatial differencing methods of the codes can be roughly equally divided into central or upwind-biased discretization. The dissipation of the central difference schemes is universally patterned after the second- and fourth-difference operators introduced by Jameson, Schmidt, and Turkel. This discretization is used in the FLO-57/FLO-67/AIRPLANE series of codes developed by Jameson, which are widely used in the aircraft industry and form the basis of many of the codes listed, such as the TEAM and TLNS3D codes. The dissipation model most frequently used is a scalar (spectral radius scaling) type with coefficient similar to those introduced originally. The coefficients have been modified in some codes to improve the shock-capturing performance at hypersonic speeds (TLNS3D), and several codes have the option of using dissipative operators designed on the basis of maintaining total-variation-diminishing (TVD) features for improving the capture of strong shocks (TEAM, MDTSL3D). The matrix dissipation technique is used in codes that also serve as Navier-Stokes solvers, such as in TLNS3D, in order to improve the resolution of the viscous layers.

The upwind-biased discretizations generally use the MUSCL approach of Van Leer. A coordinate-by-coordinate decomposition of the hyperbolic equations is used in all of the codes listed. The locally 1-D Riemann problem solved at the interface is generally accomplished with the flux-vector splittings of either Van Leer or Steger-Warming, or the flux-difference splitting of Roe. The ZEUS code solves the full, locally 1-D Riemann problem at the interface. The flux-difference-splitting approach is preferred for codes that also serve as viscous solvers because the resolution of the boundary layers is improved.

The use of shock-capturing schemes is nearly universal. Only one floating-shock-fitting code (GAUSS2D), which uses the nonconservative split-coefficient method as the basic discretization, is listed for 2-D airfoil flows. The SCRAM code is also a nonconservative discretization that uses Riemann variables, but uses a shock-fitting technique for the bow shock wave. Several other space-marching codes (SWINT, ZEUS) use shock-fitting of the bow shock wave to improve the resolution.

A hybrid discretization is used in several codes (MIM3D, F3D, LeRC-1) that combines central differencing in two directions with upwind differencing in a single direction. The hybrid differencing of MIM3D is designed to facilitate space-marching solutions by using an upwind discretization and an explicit space-marching schemes in the supersonic streamwise direction; in the crossflow direction, a central-differencing discretization with a Runge-Kutta explicit scheme with convergence acceleration is used. In the F3D and LeRC-1 codes, the upwind differencing is done in a single generalized-coordinate direction in order to facilitate the introduction of two-factor implicit schemes.

Nearly all of the codes listed are implicit schemes, if one admits that the Runge-Kutta explicit schemes with residual smooting are actually implicit schemes, because the solution to a system of algebraic equations is required to advance the solution in time. The Runge-Kutta scheme introduced by Jameson, Schmidt, and Turkel with residual smoothing and enthalpy damping is used widely in the central-difference codes listed. Other implicit codes use an approximation to the direct solution of the linear system arising from backward-time discretization; the approximations arise from spatial factorizations (CFL3D, USA, ARC3D, PARC3D, NASTD, FDL3DI), eigenvalue factorizations (EAGLE, CENS3D), or hybrid factorization relaxations (F3D, CFL3DE, EMTAC/EMTAC-MZ, LeRC-1). Several codes listed use the diagonalization method of Pulliam and Chaussee to reduce the block inversions to scalar inversions for the tridiagonal or pentadiagonal equations associated with the approximate factorizations (AF). In only two codes (ISES and FEMSAP2D) are direct solver used and both are limited to 2-D flows.

Convergence Acceleration
All of the codes use local time stepping to accelerate convergence to steady state. The multigrid technique is incorporated

in many of the central-difference Runge-Kutta codes (FLO-57, FLO-67, TLNS3D, MIM3D), patterned after the methodology of Jameson which uses the full approximation scheme (FAS). The residual smoothing, enthalpy damping, and Runge-Kutta coefficients have been designed such that convergence has been demonstrated in $O(N)$ operators for both inviscid and viscous flows. This basic approach is also used in the unstructured-grid code (NS72) developed by Mavriplis and in the FUN2D scheme. The multigrid scheme is also incorporated in the Lax-Wendroff UTRC-1 code, the MGFD code from Canada, and the implicit approximately factored CFL3D code, all of which are multiblock structured-grid codes.

Several of the codes (LaRC-1, CFL3D, EAGLE, FDL3DI, AutoFEM) offer the option of local subiterations to improve the initial approximations to the linear systems that arise from single-step factorizations or relaxations. This feature is generally used for codes that are used to solve unsteady flows. For codes that incorporate finite-rate chemistry models, the source terms that arise are treated implicitly to accelerate convergence and to overcome restrictive time-step limitations.

Special Features

The special features of the codes indicate unique or extensive application areas, flow models, boundary conditions, and/or solution algorithms. Many of the special features are highlighted in the applications section that follows. The inclusion of viscous effects is generally accomplished through incorporation of the Navier-Stokes terms. The resulting discretization uses the methodology of the baseline Euler solver to treat the convective and pressure terms; the viscous shear and heat transfer terms are treated with central differences. A number of these general-purpose codes are listed. As exceptions, the ISES code uses an integral boundary-layer model and can accommodate mildly separated flows through coupling of the integral equations with the direct Euler solution method. The UTRC-1 code uses a surface shear stress model to approximate viscous effects. Several codes that have evolved from space-marching Euler algorithms also can serve as parabolized Navier-Stokes solvers (LaRC-2, CFL3DE); two codes (USA, GASP) are sufficiently general to allow time-dependent solutions to both the Euler and Navier-Stokes equations and supersonic space-marching solutions to the Euler and parabolized Navier-Stokes equations. The codes SPARK3D, USA, GASP, SCRAM, and MDNS3D have generalized equations of state and/or finite-rate chemistry capabilities that were developed for applications to high Mach-number flows.

The majority of the codes are designed to recover steady-state solutions to the Euler equations. The ability to simulate unsteady flows, including dynamic forced-oscillation and aeroelastic coupling motions, are available in the codes F3D, CFL3D, LaRC-1, EAGLE and AutoFEM.

3.1.2 Presentation of Individual Codes

COMPANY / INSTITUTION: NASA Ames Research Center

CODE	DISCRET. TECHNIQUE	GRIDS	SPACE/TIME DISCRET.	CONVERGENCE ACCELERATION	SPECIAL FEATURES	REF.	POC
ARC3-D	3-D, finite-difference	multiblock-structured	central differencing, implicit three-factor AF scheme with diagonal or block inversions	local time stepping	Euler/Navier-Stokes solver	33	
F3D	3-D, finite-difference	multiblock-structured, patched and overset grids	hybrid: central differencing in crossflow direction and flux-vector splitting in streamwise direction, implicit two-factor AF scheme	local time stepping	Euler/Navier-Stokes solver	34	1
CENS3D	3-D, finite-difference	multiblock-structured	central differencing with flux limited dissipation, implicit two-factor LU-SGS (symmet ic Gauss-Seidel) scheme	local time stepping	Euler/Navier-Stokes solver	35	

COMPANY / INSTITUTION: Grumman Corporate Research Center

CODE	DISCRET. TECHNIQUE	GRIDS	SPACE/TIME DISCRET.	CONVERGENCE ACCELERATION	SPECIAL FEATURES	REF.	POC
MIM3D	3-D, finite-volume, cell-vertex	multiblock-structured	hybrid: upwind-differencing and implicit scheme in streamwise direction; central differencing, and Runge-Kutta explicit scheme in crossflow directions	local time stepping, residual smoothing, multigrid	conical coordinate system, supersonic flow, fighter aircraft design applications	7	2

COMPANY / INSTITUTION: Princeton University

CODE	DISCRET. TECHNIQUE	GRIDS	SPACE/TIME DISCRET.	CONVERGENCE ACCELERATION	SPECIAL FEATURES	REF.	POC
AIRPLANE	3-D, finite-element, cell-vertex	unstructured, tetrahedrons	central differencing, Runge-Kutta explicit scheme	local time stepping, residual smoothing, enthalpy damping		30, 31	3

COMPANY / INSTITUTION: Princeton University							
CODE	**DISCRET. TECHNIQUE**	**GRIDS**	**SPACE/TIME DISCRET.**	**CONVERGENCE ACCELERATION**	**SPECIAL FEATURES**	**REF.**	**POC**
FLO-67	3-D, finite-volume, cell-vertex	multiblock-structured	central differencing, Runge-Kutta explicit scheme	local time stepping, residual smoothing, enthalpy damping , multigrid	Euler/Navier-Stokes solver	8	3
FLO-57	3-D, finite-volume, cell-centered	multiblock-structured	central differencing, Runge-Kutta explicit scheme	local time stepping, residual smoothing, enthalpy damping , multigrid	Euler/Navier-Stokes solver	45, 46	

COMPANY / INSTITUTION: Lockheed Aeronautical Systems Company							
CODE	**DISCRET. TECHNIQUE**	**GRIDS**	**SPACE/TIME DISCRET.**	**CONVERGENCE ACCELERATION**	**SPECIAL FEATURES**	**REF.**	**POC**
TEAM	3-D, finite-volume, cell-vertex	multiblock-structured patched grids	central differencing scheme or upwind symmetric TVD scheme, Runge-Kutta explicit scheme	local time stepping, residual smoothing, enthalpy damping	Euler/Navier-Stokes solver, turbine applications	12, 13, 14	4

COMPANY / INSTITUTION: NASA Langley Research Center							
CODE	**DISCRET. TECHNIQUE**	**GRIDS**	**SPACE/TIME DISCRET.**	**CONVERGENCE ACCELERATION**	**SPECIAL FEATURES**	**REF.**	**POC**
USM3D	3-D, finite-volume, cell-centered	unstructured, tetrahedrons	upwind biased (flux-difference-splitting) differencing, Runge-Kutta explicit scheme	local time stepping, residual smoothing	advancing-front grid generation	18	5
CFL3D	3-D, finite-volume, cell-centered	multiblock-structured patched and/or overset grids	upwind-biased (flux-vector- or flux-difference-splitting) differencing, three-factor AF scheme with diagonal or block inversions	local time stepping, multigrid	Euler/Navier-Stokes solver, aeroelastic applications	19, 20	
NS72	3-D, finite-element, cell-vertex	unstructured, tetrahedrons, spatially adaptive	central differencing, Runge-Kutta explicit scheme	local time stepping, residual smoothing, multigrid		17	
GAUSS2D	2-D, finite-difference	monoblock-structured	upwind (split coefficient method), implicit (diagonal) AF	local time stepping	floating shock fitting, porosity boundary conditions	21	
SPARK3D	3-D, finite-difference	monoblock structured	Predictor-corrector: second-/fourth-order spatial differencing, second-order temporal differencing	local time stepping, implicit source terms	Euler/Navier-Stokes solver, finite-rate chemistry,extensive applications to combustors/nozzles	24, 52	
TLNS3D	3-D, finite-volume, cell-centered	multiblock-structured	central differencing with scalar or matrix dissipation, Runge-Kutta explicit scheme	local time stepping, residual smoothing, enthalpy damping, multigrid	Euler/Navier-Stokes solver, aeroelastic coupling applications	26, 27	
(LaRC-1)	3-D, finite-volume, cell-centered	unstructured, tetrahedrons, spatially and temporally adaptive	upwind-biased (flux-vector- or flux-difference-splitting) differencing, Runge-Kutta explicit scheme or implicit Gauss-Seidel schemes	local time stepping, residual smoothing, enthalpy damping, subiterations	aeroelastic coupling applications	22, 23, 47	
(LaRC-2)	finite-difference	monoblock-structured	upwind (flux-difference-splitting) differencing, explicit space marching	local time stepping	supersonic Euler or parabolized Navier-Stokes solver, axisymmetric option	25	
FUN2D	2-D, finite-volume, cell-vertex	unstructured, triangles, spatially adaptive	upwind-biased (flux-vector- or flux-difference-splitting) differencing, implicit red-black Gauss-Seidel schemes	local time stepping, multigrid	Euler/Navier-Stokes solver	44, 53	

COMPANY / INSTITUTION: McDonnell Aircraft Company							
CODE	**DISCRET. TECHNIQUE**	**GRIDS**	**SPACE/TIME DISCRET.**	**CONVERGENCE ACCELERATION**	**SPECIAL FEATURES**	**REF.**	**POC**
NASTD	3-D, finite-volume, cell-centered	multiblock-structured	upwind-biased (flux-difference-splitting, three-factor AF scheme	local time stepping	Euler/Navier-Stokes solver	15	6
SCRAM	3-D, finite-difference	monoblock-structured	upwind differencing with Riemann variables, explicit scheme	local time stepping	predominantly supersonic space-marching scheme, fitted bow shock, real-gas equation of state	16	
MDTSL3D	3-D, finite-volume, cell-centered	multiblock-structured	central-differencing with scalar or TVD dissipation, Runge-Kutta explicit scheme	local time stepping, residual smoothing	Euler/Navier-Stokes solver	1	
MDNS3D	3-D, finite-volume, cell-centered	multiblock-structured	central-differencing with scalar or TVD dissipation, Runge-Kutta explicit scheme	local time stepping	Euler/Navier-Stokes solver, finite-rate chemistry, extensive plume applications	2	
MDFENS	3-D, finite-element, cell-vertex-based Galerkin	unstructured, tetrahedrons	central-differencing with scalar or flux-corrected-transport dissipation, Runge-Kutta or Lax-Wendroff explicit scheme	local time stepping	Euler/Navier-Stokes solver	3	

COMPANY / INSTITUTION: Mississippi State University							
CODE	**DISCRET. TECHNIQUE**	**GRIDS**	**SPACE/TIME DISCRET.**	**CONVERGENCE ACCELERATION**	**SPECIAL FEATURES**	**REF.**	**POC**
EAGLE	3-D, finite-volume, cell-centered	multiblock-structured	upwind-biased (flux-vector- or flux-difference-splitting) differencing, two-factor LU implicit scheme	local time stepping, discrete Newton-relaxation	Euler/Navier-Stokes solver, unsteady and aeroelastic applications, propeller models, turbine applications	28, 29	7

COMPANY / INSTITUTION: Rockwell International							
CODE	**DISCRET. TECHNIQUE**	**GRIDS**	**SPACE/TIME DISCRET.**	**CONVERGENCE ACCELERATION**	**SPECIAL FEATURES**	**REF.**	**POC**
USA	3-D, finite-volume, cell-centered	multiblock-structured	upwind-biased (flux-difference-splitting) differencing, three-factor AF implicit scheme with diagonal or block inversions or explicit Runge-Kutta scheme	local time stepping	Euler/parabolized Navier-Stokes/ Navier-Stokes solver	4	8
EMTAC	3-D, finite-volume, cell-centered	monoblock-structured	upwind-biased (flux-difference-splitting) differencing, two-factor AF implicit scheme	local time stepping	predominantly supersonic Euler space-marching solver, finite rate chemistry	5	
EMTAC-MZ	3-D, finite-volume, cell-centered	multiblock-structured patched grids	upwind-biased (flux-difference-splitting) differencing, two-factor AF implicit scheme	local time stepping	predominantly supersonic Euler space-marching solver, finite rate chemistry	6	

COMPANY / INSTITUTION: United Technologies Research Center							
CODE	**DISCRET. TECHNIQUE**	**GRIDS**	**SPACE/TIME DISCRET.**	**CONVERGENCE ACCELERATION**	**SPECIAL FEATURES**	**REF.**	**POC**
(UTRC-1)	3-D, finite-volume, cell-vertex	multiblock-structured patched grids	central-differencing, Lax-Wendroff explicit scheme	local time stepping, multigrid	surface shear stress viscous model, extensive appl. to turbines & cascades	9, 10, 11	9

COMPANY / INSTITUTION: Virginia Polytechnic Institute and State University							
CODE	**DISCRET. TECHNIQUE**	**GRIDS**	**SPACE/TIME DISCRET.**	**CONVERGENCE ACCELERATION**	**SPECIAL FEATURES**	**REF.**	**POC**
GASP	3-D, finite-volume, cell-centered	multiblock-structured	upwind-biased (flux-vector- or flux-difference-splitting) differencing, explicit/implicit relaxation scheme with block or diagonal inversions	local time stepping, implicit source terms	Euler/parabolized Navier-Stokes solver, finite-rate chemistry	36, 37	10
CFL3DE	3-D, finite-volume, cell-centered	multiblock-structured patched grids	upwind-biased (flux-vector- or flux-difference-splitting) differencing, hybrid implicit relaxation scheme	local time stepping	Euler/parabolized Navier-Stokes solver	43	

COMPANY / INSTITUTION: NASA Lewis Research Center							
CODE	**DISCRET. TECHNIQUE**	**GRIDS**	**SPACE/TIME DISCRET.**	**CONVERGENCE ACCELERATION**	**SPECIAL FEATURES**	**REF.**	**POC**
(LeRC-1)	3-D, finite-difference	multiblock-structured	central-differencing, hybrid explicit/implicit scheme	local time stepping	aeroelastic and propellor applications	32	11

COMPANY / INSTITUTION: Massachusetts Institute of Technology							
CODE	**DISCRET. TECHNIQUE**	**GRIDS**	**SPACE/TIME DISCRET.**	**CONVERGENCE ACCELERATION**	**SPECIAL FEATURES**	**REF.**	**POC**
ISES	2-D, finite-volume	multiblock-structured	central-differencing, direct-solver implicit scheme	local time stepping	coupled integral boundary-layer scheme, design capability	38, 39	12

COMPANY / INSTITUTION: Naval Surface Warfare Center							
CODE	**DISCRET. TECHNIQUE**	**GRIDS**	**SPACE/TIME DISCRET.**	**CONVERGENCE ACCELERATION**	**SPECIAL FEATURES**	**REF.**	**POC**
SWINT	3-D, finite-volume	multiblock-structured	explicit MacCormack predictor-corrector scheme	local time stepping	space-marching, cylindrical coord., appl. to thin fins, fitted bow shock	40, 41	13
ZEUS	3-D finite-volume	multiblock-structured	second-order Godunov predictor-corrector scheme	local time stepping	space-marching, fitted bow shock	40, 42	

COMPANY / INSTITUTION: Wright-Patterson Air Force Base							
CODE	**DISCRET. TECHNIQUE**	**GRIDS**	**SPACE/TIME DISCRET.**	**CONVERGENCE ACCELERATION**	**SPECIAL FEATURES**	**REF.**	**POC**
FDL3DI	3-D, finite-difference	monoblock-structured	central-differencing, implicit three-factor AF with diagonal or block inversions	local time stepping, subiterations	Euler/Navier-Stokes solver	48, 49	14
FDL3DEI	3-D, finite-volume, cell-centered	monoblock-structured	upwind-biased differencing (flux-vector- or flux-difference-splitting), Runge-Kutta explicit scheme or implicit-line Gauss-Seidel scheme	local time stepping	Euler/Navier-Stokes solver	50	
WL/Leggo	3-D, finite-volume,	unstructured, hexahedrons, adaptive capability	upwind (Roe, Harten, Yee TVD), Runge-Kutta explicit scheme	local time stepping	Euler/Navier-Stokes solver	51	

COMPANY / INSTITUTION: Ecole Polytechnique / CERCA - Canada							
CODE	DISCRET. TECHNIQUE	GRIDS	SPACE/TIME DISCRET.	CONVERGENCE ACCELERATION	SPECIAL FEATURES	REF.	POC
MC3	3-D, finite-volume, cell-centered	unstructured, moving, adaptive	upwind flux-difference splitting, explicit time marching	local time stepping	moving boundaries, evolving topology, electric arc interaction	54	15

COMPANY / INSTITUTION: University of Toronto, Institute for Aerospace Studies - Canada							
CODE	DISCRET. TECHNIQUE	GRIDS	SPACE/TIME DISCRET.	CONVERGENCE ACCELERATION	SPECIAL FEATURES	REF.	POC
UTMB2D	2-D, finite-difference	multiblock-structured	central-differencing, implicit two-factor AF scheme with diagonal inversions	local time stepping	Euler/Navier-Stokes solver	55	16

COMPANY / INSTITUTION: The University of British Columbia - Canada							
CODE	DISCRET. TECHNIQUE	GRIDS	SPACE/TIME DISCRET.	CONVERGENCE ACCELERATION	SPECIAL FEATURES	REF.	POC
MGFD	finite-volume, cell-centered	multiblock-structured	power-law discretization, implicit scheme	multigrid	Euler/Navier-Stokes solver		17

COMPANY / INSTITUTION: CANADAIR - Canada							
CODE	DISCRET. TECHNIQUE	GRIDS	SPACE/TIME DISCRET.	CONVERGENCE ACCELERATION	SPECIAL FEATURES	REF.	POC
FLO67WB	3-D, finite-volume, cell-vertex	monoblock structured (C-H)	central-differencing, Runge-Kutta, explicit scheme	local time stepping, residual smoothing, enthalpy damping , multigrid	boundary-layer coupling FLO67WBV	8	18
MBTEC	3-D, finite-volume, cell-vertex	multiblock-structured (H-H), wing-body	central-differencing, Runge-Kutta, explicit scheme	local time stepping, residual smoothing, enthalpy damping, multigrid		8	

COMPANY / INSTITUTION: DE HAVILLAND - Canada							
CODE	DISCRET. TECHNIQUE	GRIDS	SPACE/TIME DISCRET.	CONVERGENCE ACCELERATION	SPECIAL FEATURES	REF.	POC
MGAERO	3-D, finite-volume, cell-centered	structured	central-differencing, Runge-Kutta, explicit scheme	local time stepping, residual smoothing, enthalpy damping, multigrid		45, 46	18

COMPANY / INSTITUTION: Concordia University - Canada							
CODE	DISCRET. TECHNIQUE	GRIDS	SPACE/TIME DISCRET.	CONVERGENCE ACCELERATION	SPECIAL FEATURES	REF.	POC
FEMSAP2D	2-D, Galerkin finite-element	monoblock unstructured, bilinear elements	steady equations, Newton linearization and direct solver	marching in artificial viscosity	Euler/Navier-Stokes solver, axisymmetric coord. option		19

COMPANY / INSTITUTION: Carleton University - Canada							
CODE	DISCRET. TECHNIQUE	GRIDS	SPACE/TIME DISCRET.	CONVERGENCE ACCELERATION	SPECIAL FEATURES	REF.	POC
AutoFEM	3-D, finite-element	unstructured, 8- or 15-node brick elements, tetrahedra or triangular elemens	Gresho's formulation with operator splitting	preconditioned conjugate gradient solver	time-accurate option, hierchical elements		20

3.1.3 Points of Contact

Point of Contact (POC) No. 1:

Code(s): ARC3D, F3D, CENS3D
Name: T. Pulliam
Dept.: Fluid Dynamics Division
Tel.: (415) 604-6417
Mailing Address: NASA Ames Research Center
MS 202A-2
Moffett Field, CA 94035-1000
Tel.-Company: (415) 604-6417
Fax-Company: (415) 604-1095
References: 33 - 35

Point of Contact (POC) No. 2:

Code(s): MIM3D
Name: F. Marconi
Dept.: Aero Science Directorate
Tel.: (516) 575-2228
Mailing Address: Grumman Corporate Research Center
MS A08-35
Bethpage, NY 11714
Tel.-Company: (516) 575-2228
Fax-Company: (516) 575-7716
References: 7

Point of Contact (POC) No. 3:

Code(s): AIRPLANE, FLO-67, FLO-57
Name: A. Jameson
Dept.: Department of Mechanical and
Aerospace Engineering
Tel.: (609) 258-5138
Mailing Address: Princeton University
Princeton, NJ 08544
Tel.-Company: (609) 258-5138
Fax-Company: (609) 258-1939
References: 8, 30, 31, 45, 46

Point of Contact (POC) No. 4:

Code(s): TEAM
Name: P. Raj
Dept.: Aerodynamics Department
Tel.: (404) 494-3801
Mailing Address: Lockheed Aeronautical Systems Company
D/73-07, Z/0685, B/L-10
Marietta, GA 30063-0685
Tel.-Company: (404) 494-3801
Fax-Company: (404) 494-3055
References: 12 - 14

Point of Contact (POC) No. 5:

Code(s): USM3D, CLF3D, NS72, GAUSS2D,
SPARK3D, TLNS3D, LaRC-1,
LaRC-2, FUN2D
Name: J. L. Thomas
Dept.: Computational Aerodynamics Branch
Tel.: (804) 864-2146
Mailing Address: NASA Langley Research Center
Mail Stop 128
Hampton, VA 23681
Tel.-Company: (804) 864-2146
Fax-Company: (804) 864-8816
References: 17 - 27, 44, 47, 52, 53

Point of Contact (POC) No. 6:

Code(s): NASTD, SCRAM, MDTSL3D, MDNS3D,
MDFENS
Name: D. Halt
Dept.: Aerodynamics and Flight Controls
Tel.: (314) 232-0519
Mailing Address: McDonnell Aircraft Company
Mail Code 111/ 1041
P. O. Box 516
St. Louis, MO 63166
Tel.-Company: (314) 232-0519
Fax-Company: (314) 777-1328
References: 1-3, 15, 16

Point of Contact (POC) No. 7:

Code(s): EAGLE
Name: D. M. Belk
Dept.: WL/MNAA
Tel.: (904) 882-3124
Mailing Address: Eglin AFB, FL 35342-5434
Tel.-Company: (904) 882-3124
Fax-Company: (904) 882-2584
References: 28, 29

Point of Contact (POC) No. 8:

Code(s): USA, EMTAC, EMTAC-MZ
Name: S. Chakravarthy
Dept. Computational Aerosciences
Tel.: (805) 373-4435
Mailing Address: Rockwell International Science Center
P. O. Box 1085
Thousand Oaks, CA 91358
Tel.-Company: (805) 373-4435
Fax-Company: (805) 373-4775
References: 4 - 6

Point of Contact (POC) No. 9:

Code(s): UTRC-1
Name: T. Barber
Dept.: Physical & Mathematical Modeling
Computational Science
Tel.: (203) 727-7619
Mailing Address: United Technologies Research Center
East Hartford, CT 06108
Tel.-Company: (203) 727-7619
Fax-Company: (203) 727-7656
References: 9 - 11

Point of Contact (POC) No. 10:

Code(s): GASP, CFD3DE
Name: R. Walters
Dept.: Department of Aerospace and Ocean
Engineering
Tel.: (703) 231-6748
Mailing Address: Virginia Polytechnic Institute and State
University
Blacksburg, VA 24060
Tel.-Company: (703) 231-6748
Fax-Company: (703) 231-9632
References: 36, 37, 43

Point of Contact (POC) No. 11:

Code(s): LeRC-1
Name: D. R. Reddy
Dept.: Computational Fluid Dynamics Branch
Tel.: (216) 433-8133
Mailing Address: NASA Lewis Research Center
Mail Stop 5-11
21000 Brookpark Road
Cleveland, OH 44135
Tel.-Company: (216) 433-8133
Fax-Company: (216) 433-8000
References: 32

Point of Contact (POC) No. 12:

Code(s): ISES
Name: M. Drela
Dept.: Aeronautics and Astronautics
Tel.: (617) 253-0067
Mailing Address: MIT Technology Licensing Office
(Heather Mapstone)
MIT E32-300
Cambridge, MA 02139
Tel.-Company: (617) 253-6966
Fax-Company: (617) 253-6790
References: 38, 39

Point of Contact (POC) No. 13:

Code(s): SWINT, ZEUS
Name: A. B. Wardlaw, Jr.
Dept.: Information and System Sciences Branch
Tel.:
Mailing Address: Naval Surface Warfare Center
Code B44
10901 New Hampshire Avenue
Silver Spring, MD 20903
Tel.-Company: (301) 394-2265
Fax-Company: (301) 394-3923
References: 40 - 42

Point of Contact (POC) No. 14:

Code(s): FDL3DI, FDL3DEI, WL/Leggo
Name: W. P. Webster
Dept.: Computational Fluid Dynamics Division
Tel.: (513) 255-2551
Mailing Address: WL/FIMC
Bldg. 450
2645 5th St., Ste. 7
Wright-Patterson AFB, OH 45433-7913
Tel.-Company: (513) 255-2551
Fax-Company: (513) 476-4210
References: 48 - 51

Point of Contact (POC) No. 15:

Code(s): MC3
Name: R. Camarero
Dept.: Centre de Recherche en Calcul Appliqué
Tel.: (514) 369-5200
Mailing Address: CERCA
5160, Boulvard Decarie
Bureau 400
Montreal, QC
CANADA H3X 2H9
Tel.-Company: (514) 369-5200
Fax-Company: (514) 369-3880
References: 54

Point of Contact (POC) No. 16:

Code(s): UTMB2D
Name: D.W. Zingg
Dept.: Institute of Aerospace Studies
Tel.: (416) 667-7709
Mailing Address: University of Toronto
4925 Dufferin Street
Downsview, Ontario
CANADA M3H 5T6
Tel.-Company: (416) 667-7709
Fax-Company: (416) 667-7799
References: 55

Point of Contact (POC) No. 17:

Code(s): MGFD
Name: M. Salcudean
Dept.: Dept. of Mechanical Engineering
Tel.: (604) 822-2781
Mailing Address: The University of British Columbia
2324 Main Mall
Vancouver, British Columbia
CANADA V6T 1Z4
Tel.-Company: (604) 822-2781
Fax-Company: (604) 822-2403
References:

Point of Contact (POC) No. 18:

Code(s): FLO67WB, MBTEC, MGAERO
Name: F. Fortin
Dept.: High Speed Aerodynamics Laboratory
Tel.: (613) 998-9201
Mailing Address: The National Research Council Canada
Institute for Aerospace Research
CANADA K1A 0R6
Tel.-Company: (613) 998-9201
Fax-Company: (613) 998-1281
References: 8, 45, 46

Point of Contact (POC) No. 19:

Code(s): FEMSAP2D
Name: W.G. Habashi
Dept.: Dept. of Mechanical Engineering
Tel.: (514) 848-3165
Mailing Address: Concordia University CFD Lab
1455 De Maisonneuve W.
Montreal, Quebec
CANADA H3G 1M8
Tel.-Company: (514) 848-3165
Fax-Company: (514) 848-8601
References:

Point of Contact (POC) No. 20:

Code(s): AutoFEM
Name: J.A. Goldak
Dept.: Dept. of Mechanical & Aerospace Engineering
Tel.: (613) 788-5688
Mailing Address: Carleton University
1125 Colonel By Drive
Ottawa, Ontario
CANADA K1S 5B6
Tel.-Company: (613) 788-5688
Fax-Company: (613) 788-8958
References:

3.1.4 References

1. DEESE, J. and AGARWAL, R. "Navier-Stokes Calculations of Transonic Viscous Flow about Wing/Body Configurations," *Journal of Aircraft*, Vol. 25, No. 12, Dec. 1988, pp. 1106-1112

2. GIELDA, T., DEESE, J., and AGARWAL, R. "Three-Dimensional Delta/Thor Launch Vehicle Analysis Including Main Engine Plume Effects," AIAA 91-3338-CP, Sept. 1991.

3. MARCUM, D. L. and AGARWAL, R. K. "A Three-Dimensional Finite Element Navier-Stokes Solver with Two-Equation Turbulence Model for Unstructured Grids," AIAA 90-1652, June 1990.

4. CHAKRAVARTHY, S. R., SZEMA,K.Y. and HANEY, J. W. "Unified 'Nose-to-Tail' Computational Method for Hypersonic Vehicle Applications," AIAA 88-2564, June 1988.

5. CHAKRAVARTHY, S. R. and SZEMA, K. Y. "An Euler Solver for Three-Dimensional Supersonic Flows with Subsonic Pockets," *Journal of Aircraft*, Vol. 24, (1987), pp.73-83.

6. SZEMA, K. Y., CHAKRAVARTHY, S. R., and SHANKAR, V. "Supersonic Flow Computations Over Aerospace Configurations Using an Euler Maching Solver," NASA CR-4085, July 1987.

7. SICLARI, M. and DARDEN, C. "CFD Predictions of the Near-Field Sonic Boom Environment for Two Low Boom HSCT Configurations," AIAA 91-1631, June 1991.

8. VOLPE, G. and JAMESON, A. "Efficient Method for Computing Transonic and Supersonic Flows about Aircraft," Journal of Aircraft, Vol. 27, No. 3, March 1990, pp. 223-231.

9. NI, R. H. "A Multiple-Grid Scheme for Solving the Euler Equations," AIAA 81-1025, June 1981.

10. NI, R. H. and BOGOIAN, J. C. "Prediction of 3D Multi-Stage Turbine Flow Field Using a Multiple-Grid Euler Solver," AIAA 89-0203, 1989.

11. NI, R. H., SHARMA, O. P., TAKAHASHI, R. K., and BOGOIAN, J. C. "3D Unsteady Flow Simulation Through a Turbine Stage," Aeronautical Engineering Conference, Melbourne, Australia, Oct. 9-11, 1989.

12. RAJ, P., OLLING, CR., and SINGER, S. W. "Applications of Multizone Euler/Navier-Stokes Aerodynamics Methods to Aircraft Configurations," ICAS 90-6.4.4, 1990.

13. SINGER, S. W. and MATTSON, E. A. "Internal and External Flow Simulations Using Multizone Euler/Navier-Stokes Aerodynamic Methods," SAE Technical Paper Series No. 901856, October 1990.

14. RAJ, P. and SINGER, S. W. "Computational Aerodynamics in Aircraft Design: Challenges and Opportunities for Euler/Navier-Stokes Methods," iPAC 911990, October 1991.

15. HINDASH, I. O., BUSH, R. H., and COSNER, R. R. "Computational Modeling of Inlet Hammershock Wave Generation," AIAA 90-2005, July 1990.

16. VERHOFF, A., STROOKESBERRY, D. C., HOPPING, B. M., and MICHAL, T. R. "Supersonic/Hypersonic Euler Flowfield Prediction Method for Aircraft Configurations," Fourth Symposium on Numerical & Physical Aspects of Aerodynamic Flows, Long Beach, CA, Jan. 1989.

17. MAVRIPLIS, D. J. "Three-Dimensional Unstructured-Multigrid for the Euler Equations," AIAA 91-1549-CP, June 1991.

18. FRINK, N. T., PARIKH, P., and PIRZADEH, S. "Aerodynamic Analysis of Complex Configurations Using Unstructured-Grids," AIAA 91-3292, Sept. 1991.

19. ANDERSON, W. K., THOMAS, J. L., and RUMSEY, C. L. "Extension and Application of Flux-Vector Splitting to Calculations on Dynamic Meshes," *AIAA Journal*, Vol. 27, No. 6, June 1989, pp. 773-774.

20. THOMAS, J. L., TAYLOR, S. L., and ANDERSON, W. K. "Navier-Stokes Computations of Vortical Flows Over Low Aspect Ratio Wings," *AIAA Journal*, Vol. 28, No. 2, Feb. 1990, pp. 205-212.

21. HARTWICH, P. M. "Fresh Look at Floating Shock Fitting," AIAA Journal, Vol. 29, No. 7, July 1991, pp. 1084-1091.

22. BATINA, J. T. "Implicit Flux-Split Euler Schemes for Unsteady Aerodynamic Analysis Involving Unstructured Dynamic Meshes," AIAA 90-0936, April 1990.

23. BATINA, J. T. "Unsteady Euler Algorithm with Unstructured Dynamic Mesh for Complex-Aircraft Aeroelastic Analysis," AIAA 89-1189, April 1989.

24. CARPENTER, M. H. "Three-Dimensional Computations of Cross-Flow Injection and Combustion in a Supersonic Flow," AIAA Paper 89-1870, June 1989.

25. KORTE, J. J. "An Explicit Upwind Algorithm for Solving the Parabolized Navier-Stokes Equations," NASA TP 3050, Feb. 1991.

26. SWANSON, R. C. and TURKEL, E. "Artificial Dissipation and Central Difference Schemes for the Euler and Navier-Stokes Equations," AIAA 87-1107, June 1987.

27. VATSA, V. N. "Accurate Solutions for Transonic Viscous Flow Over Finite Wings," *Journal of Aircraft*, Vol. 24, No. 1, Jan. 1987, pp. 377-385.

28. WHITFIELD, D. L. and TAYLOR, L. K. "Discretized Newton-Relaxation Solution of High Resolution Flux-Difference Split Schemes," AIAA 91-1539, June 1991.

29. BELK, D. M. and WHITFIELD, D. L. "Time-Accurate Euler Equations Solutions on Dynamic Blocked Grids," AIAA 87-1127-CP, June 1987.

30. JAMESON, A., BAKER T. J., and WEATHERILL, N. P. "Calculation of Inviscid Transonic Flow Over a Complete Aircraft, AIAA 86-0103, Jan. 1986.

31. JAMESON, A. and BAKER, T. J. "Improvements to the Aircraft Euler Method," AIAA 87-0452, Jan. 1987.

32. SRIVASTAVA, R., SANKAR, N. L., REDDY, T. S. R., and HUFF, D. L. "Application of an Efficient Hybrid Scheme for Aeroelastic Analysis of Advanced Propellers," AIAA 90-0028, Jan. 1990.

33. PULLIAM, T. H. and STEGER, J. L. "Implicit Finite-Difference Simulation of Three-Dimensional Compressible Flow," *AIAA Journal*, Vol. 18, No. 2, 1980, pp. 159-167.

34. YING, S. X., STEGER, J. L., SCHIFF, L. B., and BAGANOFF, D. "Numerical Simulation of Unsteady, Viscous, High-Angle-of-Attack Flows Using a Partially Flux-Split Algorithm," AIAA 86-2179, Aug. 1986.

35. YOON, S. and KWAK, D. "An Implicit Three-Dimensional Navier-Stokes Solver for Compressible Flow," AIAA 91-1555-CP, June 1991.

36. GROSSMAN, B. and CINELLA, P. "Flux-Split Algorithms for Flows with Non-Equilibrium Chemistry and Vibrational Relaxation," *Journal of Computational Physics*, Vol. 88, No. 1, 1990.

37. WALTERS, R. W., CINELLA, P., SLACK, D. C. and HALT, D. "Characteristic Based Algorithms for Flows in Thermo-Chemical Non-Equilibrium," AIAA 90-0393, 1990.

38. DRELA, M. "Two-Dimensional Transonic Aerodynamic Design and Analysis Using the Euler Equations," Ph.D. Dissertation, Department of Aeronautics and Astronautics, Massachusetts Institute of Technology, Dec. 1985.

39. GILES, M. B. "Newton Solution of Steady Two-Dimensional Transonic Flow," Ph. D. dissertation, Department of Aeronautics and Astronautics, Massachusetts Institute of Technology, June 1985.

40. WARDLAW, A. B., JR., BALTAKIS, F. P., PRIOLO, F. J., and SOLOMON, J. M. "Space Marching Euler Solvers," in Tactical Missile Aerodynamics: Prediction Methodology, AIAA Progress in Astronautics and Aeronautics, Vol. 142, May 1991.

41. WARDLAW, A. B., JR., BALTAAKIS, F. P., SOLOMON, J. M., and HACKERMAN, L. B. "An Inviscid Computational Method for Tactical Missile Configurations," Naval Surface Warfare Center, TR-81-457, Silver Spring, MD, December 1981.

42. WARDLAW, A. B., JR., and DAVIS, S. F. "A Second Godunov's Method for Tactical Missiles," Naval Surface Warfare Center, TR-86-506, Silver Spring, MD, April 1987.

43. WALTERS, R. W., THOMAS, J. L., and VAN LEER, B. "An Implicit Flux-Split Algorithm for the Compressible Euler and Navier-Stokes Equations," *Lecture Notes in Physics*, Vol. 264, Springer-Verlag, 1986, pp. 628-635.

44. ANDERSON, W. K. "A Grid Generation and Flow Solution Method for the Euler Equations on Unstructured Grids," NASA TM-4295, April 1992.

45. JAMESON, A., SCHMIDT, W., and TURKEL, E. "Numerical Solutions for the Euler Equations by Finite-Volume Methods Using Runge-Kutta Time-Stepping Schemes," AIAA Paper 81-1259, June 1981.

46. JAMESON, A. and BAKER, T. J. "Multigrid Solution of the Euler Equations for Aircraft Configurations," AIAA 85-0093, Jan. 1985.

47. RAUSCH, R. D., BATINA, J. T., and YANG, H. T. Y. "Spatial Adaption Procedures on Tetrahedral Meshes for Unsteady Aerodynamic Flow Calculations," AIAA Paper 93-0670, January 1993.

48. VISBAL, M. R. "Structure of Vortex Breakdown on a Pitching Delta Wing," AIAA Paper 93-0434, 1993.

49. RIZZETTA, D. P. and VISBAL, M. R. "Comparative Numerical Study of Two Turbulence Models for Airfoil Static Dynamic Stall," AIAA Paper 92-4649-CP, 1992.

50. GAITONDE, D. V. and SHANG, J. S. "Numerical Simulation of Flow Past the X24C Reentry Vehicle," AIAA Paper 93-0319, 1993.

51. AFTOSMIS, M. J. "Viscous Flow Simulation Using and Upwind Method for Hexahedral Adaptive Meshes," AIAA Paper 93-0772, 1993.

52. DRUMMOND, J. P. "Supersonic Reacting Internal Flow Fields: Numerical Approaches to Combustion Modeling," Chapter 12, (ed. E. Oran and J. Boris), American Institute of Astronautics and Aeronautics, Washington D. C., 1991. Also NASA-TM-103480, 1992.

53. ANDERSON, W. K. and BONHAUS, D. "An Implicit Upwind Algorithm for Computing Turbulent Flows on Unstructured-Grids," *Computers and Fluids*, Vol.23, No. 1, 1994, pp. 1-21.

54. TREPANIER, J. REGGIO, M., PARASCHIVOIU, M. and CAMARERO, R. "Unsteady Euler Solutions for Arbitrarily Moving Bodies and Boundaries," AIAA-Paper 92-0051, 1992.

55. NELSON, T. E., ZINGG, D. W. and JOHNSTON, G. W. "Numerical Solutions of the Navier-Stokes Equations for High-Lift Configurations on Structured Composite Grids," Proc. of the 71st AGARD FDP-Meeting on High-Lift Aerodynamics, Banff, Canada, Oct. 5-8, 1992.

3.2 CODES FROM FRANCE, BELGIUM, ITALY GREECE AND TURKEY

3.2.1 OVERVIEW

Among the 3-D Euler codes used in industry for complex configurations, a majority of them is of the multiblock structured grid type and practically all codes are derived from a finite-volume discretization. These codes, developed either directly by industrial companies or by research institutes, are generally implemented together with an independent grid generation system.

It is noted that in France and in Belgium upwind schemes have been largely favoured for many years and that, for the case of space centered differencing, classical variants of the Lax Wendroff scheme (SESAME, AEROLOG) or a more original one (WAVES) have been preferred over the Runge-Kutta central schemes of Jameson et al (e.g. Ref. 3.0,1). By contrast Jameson schemes are favoured in the 3-D codes used by industry in Italy.

However, the tendency for most of multi-application codes has been to progressively incorporate variants which are added to existing capabilities in order to provide the users with a larger choice according to their needs. Indeed, the different objectives of robustness, accuracy or low computer costs can strongly influence the choice of the options available in a general code for various applications.

Another common tendency is the extension of Euler codes to Navier-Stokes solvers by including viscous terms and turbulence models.

After these general statements we will survey the various codes developed to solve the Euler equations by different companies or research laboratories.

In **France** at ONERA (French National Aerospace Research Institute) several different Euler codes have been worked out. The SESAME code is a multidomain[1,2] solver with or without overlapping of blocks based on a Lax-Wendroff-Ni finite-volume method of cell vertex type with a multigrid acceleration technique.,[3,4] It has been recently completed with a Runge-Kutta time stepping scheme in a cell-centered discretization making the multiblock boundary treatment easier. Both internal (turbomachinery[3-6]) and external (transport aircraft[7]) flows can be computed with this code which has a closely related Navier-Stokes extension (CANARI). The WAVES code is more specific and it is founded on the implicit Lerat scheme which has the unique feature of being a centered scheme without any added artificial viscosity.[9-11] Its domain of application has been mainly helicopter rotor flows.[8] Its fusion with the SESAME code has been recently achieved.

Two other codes have reached a level of industrial use, both in the class of upwind methods. Firstly, the FLU3C code was developed in close cooperation with the Missiles Division of AEROSPATIALE[22] specially for supersonic applications.[23-26] Mainly developed at ONERA and much more general with its multiblock capability and its large number of options (various numerical fluxes in a MUSCL approach, different multizone techniques, implicit time stepping, Navier-Stokes extensions) and its efficiency particularly for supersonic and hypersonic flows (space-marching, real gas effects), the FLU3M code[13-21] has a structure which allows both complex applications and the implementation of new modules at the research or development level. Other Euler codes can be quoted which are developed at ONERA in the Structures Department for aeroelasticity (EF3D[27-28]) and in the Propulsion Department.[29-30]

AEROSPATIALE uses Euler codes provided by ONERA (FLU3C, SESAME, FLU3M) but has also made some development of codes such as CELGR3D in the Space and Defence Division. This solver is of unstructured type with hexahedra and upwind schemes and finds its domain of application in supersonic or hypersonic flows.[32-33]

A Lax-Wendroff-Ni scheme was the basis of the development of a multiblock structured grid code, the AEROLOG code at MATRA DEFENSE[34-35]. This code with acceleration by multigrid and implicit residual smoothing is used for steady or unsteady flows past missiles.

The Euler code EUGENIE developed at DASSAULT AVIATION is the result of joint research studies[37-46] with INRIA on unstructured grid methods and upwind schemes in a finite-volume node centered formulation. Multigrid and implicit approach are combined in order to improve convergence. Mesh adaption either by displacement or refinement and hypersonic capabilities are available. It is worth to mention here the strong impetus given through Europe to the research on hypersonic flow solvers through the Hermes programme and the wokshops on hypersonic flows for reentry problems held at INRIA.[18,19,31,32,51,66,70,72]

Two research centers have been concerned in **Belgium** with the development of CFD codes, namely the von Karman Institute (VKI) and the Free University of Brussels (VUB).

Many 2-D research codes were developed at VKI both for unstructured[51-54] and structured[55-58] grids with a 3-D Euler code M3D[55-57] containing several options in particular with different upwind and various time stepping schemes. A noteworthy activity at VKI is the development of genuinely multidimensional upwind schemes[53-54].

The EURANUS[48-50] code developed at VUB in cooperation with FFA for ESA appears as a general software system capable of solving both Euler and Reynolds-Averaged Navier-Stokes equations. This structured multiblock code includes both upwind TVD and central Jameson schemes with the FAS multigrid method. Special care has been paid to the data management in order to give flexibility in the choice of the computer used.

In **Italy**, industrial codes are developed by ALENIA in the Torino and Naples centers. In ALENIA Defence Aircraft Division, the UES3D[59-62] code is based on a generalization of a central differencing scheme of Jameson type for unstructured grids, using a node-centered finite-volume approach. Explicit Runge Kutta time stepping is used with convergence acceleration by local time stepping, residual averaging and enthalpy damping. Grid generation is done with the unstructured mesh generator M3DU giving the code the capability to calculate a complete aircraft. Another 3-D code is in use at ALENIA DAD which is of multiblock structured type. This code (MES3D[62-63]) contains a cell-centered scheme with central differencing and it is built with the same options as those described above for UES3D.

A code in use at ALENIA GAT is ENSOLV which has been developed in cooperation with NLR. This code is described in section 3.4 and offers features rather similar to those mentioned for MES3D.

CIRA has also a 3-D Euler code ETF3D, a monoblock structured grid solver with Jameson type scheme, dedicated to the calculation of transonic wing flows. The 2-D multiblock version of this code has been coupled with an integral boundary layer method in direct or inverse mode[64-65]. Several other 2-D solvers have been developed upon upwind schemes with the Borelli/Pandolfi Rieman-solver in a finite-volume cell-centered formulation. These research codes are aimed at solving supersonic or hypersonic flows with real gas effects.[66-70]

In the same line of studies for hypersonic problems it is worth mentioning the research codes developed at the University of Roma. These 2-D Euler codes are based on upwind schemes of different types, a generalization of the Moretti scheme with application to reactive flows[71-74] a hybrid of non-conservative and conservative schemes[75-76] and a Godunov-type scheme in predictor-corrector formulation.[77]

In **Greece** at the National Technical University of Athens (NTUA) several Euler codes have been developed aiming on

external as well as internal (turbomachinery applications) aerodynamic flow problems.

The 2-D single-block, structured finite-difference EU2D[78-80] code is mainly used for investigations of unsteady, internal and external flow problems in the range from the subsonic to supersonic regime. The code is based on a central discretization scheme including an artificial dissipation operator providing the necessary upwind bias. Both, time and space discretization are of 2nd order accuracy. An alternating direction implicit (ADI) procedure is used for time integration. The code has a moving mesh capability to treat appropriate unsteady problems.

As a further development the 2-D multiblock finite-volume code NSWIND[81-85] is mentioned. The code has various discretization options like a modified upwind flux vector splitting as well as the flux difference splitting method including the Godunov-type Riemann-solver approach of Eberle (see also Ref. 3.3,22-31). By a multi-level mesh sequencing and multigrid technique convergence is accelerated to steady state. A composite zonal solution technique, in which for various flow regions different governing equations are solved is also noted. The code is also able to operate in time-accurate mode. The implicit operator is inverted by an unfactored method using Gauss-Seidel relaxation for solution.

A 2-D and 3-D multiblock finite-difference code ATHENA[86] essentially based on Jameson's central-differenced operator for convective terms including the blended second and fourth difference dissipation operator is applied mainly for turbomachinery flows. For time integration an explicit fractional step or an implicit ADI method may be used. The code has also a Navier-Stokes capability.

In **Turkey** a 3-D, cell-vertex, finite-volume code, called ER3D[87], is developed at the ROKETSAN company, which is based on a hexahedral elementary control volume and an unstructured data structure to enhance flexibility. Time stepping is based on second order accurate Lax-Wendroff scheme. Applications are covering the subsonic to supersonic flow regime.

3.2.2 Presentation of Individual Codes

COMPANY / INSTITUTION: ONERA - France							
CODE	**DISCRET. TECHNIQUE**	**GRIDS**	**SPACE/TIME DISCRET.**	**CONVERGENCE ACCELERATION**	**SPECIAL FEATURES**	**REF.**	**POC**
SESAME	3-D, finite-volume, cell-vertex or cell-centered	multiblock-structured, patched or overlapping blocks	space-centered with either Lax-Wendroff-Ni or Runge-Kutta time stepping scheme	local time stepping, multigrid (Ni-method) or implicit residual smoothing (Lerat)	Navier-Stokes extension: CANARI code	1-7	
WAVES	3-D, finite-volume, cell-centered	monoblock-structured	space-centered with Lax-Wendroff-type (Lerat extension) , implicit stage with ADI factorization	local time stepping	no addition of artificial viscosity	8-12	1
FLU3M	3-D, finite-volume, node centered or cell-centered	multiblock-structured overlapping subdomains (Chimera technique)	upwind explicit/implicit MUSCL schemes (van Leer, Roe, Osher, Approximate Riemann solvers), block implicit ADI factorization scheme	local time stepping	Euler/Navier-Stokes solver, space marching capability, equilibrium real gas, two species gas	13-21	

COMPANY / INSTITUTION: ONERA & Aerospatiale - France							
CODE	**DISCRET. TECHNIQUE**	**GRIDS**	**SPACE/TIME DISCRET.**	**CONVERGENCE ACCELERATION**	**SPECIAL FEATURES**	**REF.**	**POC**
FLU3C	3-D, finite-volume, node centered	monoblock, structured	second order upwind explicit MUSCL scheme with van Leer flux vector splitting	local time stepping	space marching capability for supersonic flows, *industrial application:* missiles, launchers, etc.	22-26	1

COMPANY / INSTITUTION: Aerospatiale - France							
CODE	**DISCRET. TECHNIQUE**	**GRIDS**	**SPACE/TIME DISCRET.**	**CONVERGENCE ACCELERATION**	**SPECIAL FEATURES**	**REF.**	**POC**
CEL3GR	3-D, finite-volume, cell-centered	unstructured hexahedra	MUSCL scheme with Sanders-Prendergast flux splitting scheme or Osher Riemann solver, time stepping by two step Runge-Kutta or linearized implicit Jacobi relaxation	local time stepping	equilibrium real gas option	32-33	2

COMPANY / INSTITUTION: Matra Defense - France							
CODE	DISCRET. TECHNIQUE	GRIDS	SPACE/TIME DISCRET.	CONVERGENCE ACCELERATION	SPECIAL FEATURES	REF.	POC
AEROLOG	3-D, finite-volume, cell-vertex	multiblock-structured	Lax-Wendroff one step explicit scheme, steady or time accurate mode	multigrid (Ni-method), implicit residual smoothing (Lerat)	steady or time accurate mode, inertial or non-inertial frame of reference, two species capability	34-35	3

COMPANY / INSTITUTION: Dassault Aviation - France							
CODE	DISCRET. TECHNIQUE	GRIDS	SPACE/TIME DISCRET.	CONVERGENCE ACCELERATION	SPECIAL FEATURES	REF.	POC
EUGENIE	3-D, Galerkin formulation, finite-volume, node centered	unstructured multielement, mesh refinement and mesh deformation capability	upwind MUSCL extension, generalized implicitly linearized Osher Riemann-solver	multigrid	reactive flow simulation option, edge based data structure	36-47	4

COMPANY / INSTITUTION: Vrije Universiteit Brussel - Belgium							
CODE	DISCRET. TECHNIQUE	GRIDS	SPACE/TIME DISCRET.	CONVERGENCE ACCELERATION	SPECIAL FEATURES	REF.	POC
EURANUS	3-D, finite-volume, cell-centered	multiblock-structured	Roe upwind TVD, Yee symmetric TVD or central Jameson scheme with blended second and fourth difference dissipation operator, explicit Runge-Kutta and SOR / SLOR implicit relaxation time stepping schemes	local time stepping, multigrid (FAS), implicit residual smoothing for Runge-Kutta	Euler/Navier-Stokes solver, node coincidence at block interfaces	48-50	5

COMPANY / INSTITUTION: VKI (Von Karman Institute for Fluid Dynamics) - Belgium							
CODE	DISCRET. TECHNIQUE	GRIDS	SPACE/TIME DISCRET.	CONVERGENCE ACCELERATION	SPECIAL FEATURES	REF.	POC
M3D	3-D, finite-volume, cell-centered	monoblock-structured	Roe flux difference splitting, van Leer flux vector splittting upwind and MUSCL TVD scheme, explicit Euler forward, explicit Runge-Kutta and implicit Euler backward relaxation time stepping scheme	local time stepping, explicit and implicit residual smoothing	shock capturing, subsonic to hypersonic (cold) flow	55-57	6

COMPANY / INSTITUTION: Alenia Aeronautica, DAD - Italy							
CODE	DISCRET. TECHNIQUE	GRIDS	SPACE/TIME DISCRET.	CONVERGENCE ACCELERATION	SPECIAL FEATURES	REF.	POC
UES3D	3-D, finite-volume, node-centered	unstructured, local mesh refinement	central Jameson scheme with blended second and fourth difference dissipation, explicit Runge-Kutta time stepping scheme	local time stepping, residual smoothing, enthaly damping		59-62	7
MES3D	3-D, finite-volume, cell-centered	multiblock-structured	central Jameson scheme with blended second and fourth difference dissipation, explicit Runge-Kutta time stepping scheme	local time stepping, residual smoothing, enthaly damping		62-63	

COMPANY / INSTITUTION: Alenia Aeronautica G.A.T. - Italy							
CODE	DISCRET. TECHNIQUE	GRIDS	SPACE/TIME DISCRET.	CONVERGENCE ACCELERATION	SPECIAL FEATURES	REF.	POC
ENSOLV	3-D, finite-volume, cell-centered	multiblock-structured block-to-block mesh refinement and adaptation	central Jameson scheme with blended second and fourth difference dissipation, explicit multistage Runge-Kutta time stepping scheme	local time stepping, implicit residual averaging, enthalpy damping, multigrid - full & semi-coarsening	Euler/Navier-Stokes solver, part of CFD system ENFLOW (see section 3.4)	in sect 3.4 21-41	8

COMPANY / INSTITUTION: C.I.R.A. - Italy							
CODE	DISCRET. TECHNIQUE	GRIDS	SPACE/TIME DISCRET.	CONVERGENCE ACCELERATION	SPECIAL FEATURES	REF.	POC
ETF3D	3-D, finite-volume, cell-centered	monoblock-structured	central Jameson scheme with blended second and fourth difference dissipation, explicit Runge-Kutta time stepping scheme	FAS multigrid		64-65	9

COMPANY / INSTITUTION: National Technical University of Athens (NTUA) - GREECE							
CODE	DISCRET. TECHNIQUE	GRIDS	SPACE/TIME DISCRET.	CONVERGENCE ACCELERATION	SPECIAL FEATURES	REF.	POC
EU2D	2-D, axisymm., finite-difference	single block-structured	second order central discretization, artificial dissipation scheme, approximate factorization procedure, ADI, second order time accuracy		time accurate code, moving mesh capability	78-80	10
NSWIND	2-D, axisymm., finite-volume, cell-centered	multiblock-structured, block-by-block grid refinement,	upwind flux vector (FVS) splitting and flux difference splitting (FDS) discretization, Eberle's Riemann-solver, second order MUSCL scheme, implicit unfactored Gauss-Seidel relaxation	local time stepping, multi-level procedure, multigrid	time accurate option, zonal methods capability, real gas option	81-85	
ATHENA	2-D, 3-D, and quasi 3-D, finite-difference	multiblock-structured	central Jameson scheme with blended second and fourth difference dissipation, explicit fractional step and implicit ADI time integration scheme	local time stepping	Euler/Navier-Stokes solver, quasi 2-D streamtube turbo-machinery option	86	

3.2.3 Points of Contact

Point of Contact (POC) No. 1:

Code(s): SESAME, WAVES, FLU3M, FLU3C
Name: J.P. Veuillot
Dept.: Aerodynamics Department
Tel.: (+33) 1 46 73 42 68
Mailing Address: ONERA
BP 72
F-92322 Châtillon Cedex
France
Tel.-Company: (+33) 1 46 73 40 40
Fax-Company: (+33) 1 46 73 41 41
References: 1 - 26

Point of Contact (POC) No. 2:

Code(s): CEL3GR
Name: F. Dubois
Dept.: Applied Mathematics Department
Tel.: (+33) 1 34 92 28 57
Mailing Address: AEROSPATIALE Espace et Defense
BP 2
F-78133 Les Mureaux Cedex
France
Tel.-Company: (+33) 1 34 92 12 34
Fax-Company: (+33) 1 34 92 39 15
References: 32 - 33

Point of Contact (POC) No. 3:

Code(s): AEROLOG
Name: M. Bredif
Dept.: Aerodynamics Department
Tel.: (+33) 1 34 88 37 47
Mailing Address. MATRA DEFENSE
37 Avenue Louis Breguet
F-78146 Velizy Cedex
France
Tel.-Company: (+33) 1 34 88 30 00
Fax-Company: (+33) 1 34 65 12 15
References: 34, 35

Point of Contact (POC) No. 4:

Code(s): EUGENIE
Name: B. Stoufflet
Dept.: Aerodynamics Department
Tel.: (+33) 1 47 11 34 22
Mailing Address: DASSAULT AVIATION
78 Quai Marcel Dassault
F-92214 Saint Cloud
France
Tel.-Company: (+33) 1 47 11 40 00
Fax-Company: (+33) 1 47 11 49 01
References: 36 - 47

Point of Contact (POC) No. 5:

Code(s): EURANUS
Name: C. Lacor
Dept.: Computational Fluid Dynamics Group
Tel.: (+32) 2 641 23 79
Mailing Address: Vrije Universiteit Brussel (VUB)
Pleinlaan 2
B-1050 Brussel
Belgium
Tel.-Company: (+32) 2 641 23 91
Fax-Company: (+32) 2 641 28 80
References: 48 - 50

Point of Contact (POC) No. 6:

Code(s): M3D
Name: H. Deconinck
Dept.: Computational Fluid Dynamics Group
Tel.: (+32) 2 358 19 01 ext 237
Mailing Address: Von Karman Inst. for Fluid Dynamics (VKI)
72 Chaussee de Waterloo
B-1640 Rhode - St - Genese
Belgium
Tel.-Company: (+32) 2 358 19 01
Fax-Company: (+32) 2 358 28 85
References: 55 - 57

Point of Contact (POC) No. 7:

Code(s): MES3D, UES3D
Name. M. Borsi
Dept.: Defence Aircraft Division
Tel.: (+39) 11 718 1017
Mailing Address: ALENIA
Corso Marche 41
I-10146 Torino
Tel.-Company: (+39)11 718 1789
Fax-Company: (+39)11 718 1078
References: 59 - 63

Point of Contact (POC) No. 8:

Code(s): ESOLVB
Name: P.L.Vitagliano
Dept.: G.A.T.
Tel.: (+39) 081 845 3459
Mailing Address: ALENIA
Viale dell'Aeronautica
I-80038 Pomgliano d'Arco (NA)
Tel.-Company: (+39) 081 845 3459
Fax-Company: (+39) 081 845 2142
References: (see sect. 3.4, ref. 21 - 41)

Point of Contact (POC) No. 9:

Code(s): FTF3D
Name: P. de Matteis
Dept.: Aerodynamics
Tel.: (+39) 823 623311
Mailing Address: C.I.R.A.
Via Maiorise
I-81043 Capua
Tel.-Company: (+39) 823 623111
Fax-Company: (+39) 823 622060
References: 64, 65

Point of Contact (POC) No. 10:

Code(s): EU2D, NSWIND, ATHENA
Name: S. Tsangaris
Dept.: Aerodynamic Laboratory
Tel.: (+30) 1-77-13060
Mailing Address: National University of Athens (NTUA)
P.O. Box 64070
157 10 Zografou
Greece
Tel.-Company: (+30) 1-77-13060
Fax-Company: (+30) 1-77-06545 or 1-77-84582
References: 78 - 86

3.2.4 References

1. VIVIAND, H. and VEUILLOT, J.P., "Méthodes pseudo-instationnaires pour le calcul d'écoulements transsoniques," ONERA Publication Nr. 1978-4 and English translation "Pseudo-Unsteady Methods for the Computation of Transonic Flows," ESA TT 561.

2. CAMBIER, L., DUSSON, V. and VEUILLOT, J.P., "Multidomain Methods for Solution of the Euler Equations. Applications with Overlapping Subdomains," *La Recherche Aerospatiale*, Vol. 1985-3, pp. 181-188 (French and English editions).

3. COUAILLIER, V., "Solution of the Euler Equations: Explicit Scheme Acceleration by a Multigrid Method," Second European Conference on Multigrid Methods, GAMM Cologne, Oct. 1985, GMD-Studien Nr. 110 and TP ONERA Nr. 1985-129.

4. COUAILLIER, V. and PEYRET, R., "Theoretical and Numerical Study of the Ni's Multigrid Method," La Recherche Aerospatiale, Nr. 1985-5, (French and English editions).

5. VUILLOT, A.M., "A Multi-Domain 3D Euler Solver for Flows in Turbomachines," Proceedings of the 9th ISABE Symposium, Athens, Sept. 1989.

6. COUAILLIER, V., "Multigrid Methods for Solving Euler and Navier-Stokes Equations in Two and Three Dimensions," Proceedings of the 8th GAMM Conf. on Numerical Methods in Fluid Dynamics, Notes on Numerical Fluid Mechnics, Vol. 29, Pieter Wesseling Ed., Vieweg 1990

7. COUAILLIER, V., LIAMIS, N. and GRENON, R. "Transonic and Supersonic Flow Calculations around Aircrafts Using a Multidomain Euler Code," Proceedings of the 13th ICNMFD, Lecture Notes in Physics, Springer-Verlag, Berlin Heidelberg 1993.

8. SIDES, J. and BONIFACE, J.C. "Solution of the Compressible Euler Equations for Steady Flows around Helicopter Rotor Blades by an Implicit Space-Centered Method," Proceedings of the 17th Rotorcraft Forum, Berlin, September 23-27, 1991. Also TP ONERA Nr. 1991-177.

9. LERAT, A. "Implicit Methods of Second Order of Accuracy for the Euler Equations," AIAA Paper 83-1925. Also *AIAA Journal* Vol. 23 (1985), pp. 33-40.

10. LERAT, A., SIDES, J. and DARU, V. "An Implicit Finite Volume Method for Solving the Euler Equations," Lecture Notes in Physics Vol. 170, 1982, pp. 343-349.

11. LERAT, A. and SIDES, J. "Efficient Solution of the Steady Euler Equations with a Centered Implicit Method," In: Numer. Meth. Fluid Dyn. III, Morton K.W. and Baines M.J. Eds., Clarendon Press, Oxford, pp. 65-86, 1988. Also TP ONERA Nr. 1988-128.

12. LERAT, A. and SIDES, J. "Implicit Transonic Calculations without Artificial Viscosity or Upwinding," Proceedings of GAMM - Workshop on the Numerical Solution of the Compressible Euler Flows," Notes on Numerical Fluid Mechanics Vol. 26, 1989, pp. 227-250.

13. GUILLEN, Ph. and DORMIEUX, M. "Design of a Multidomain Euler Code," Computational Mechanics Institute, "Supercomputing in Fluid Flow" Conference, Boston, Oct. 1989.

14. BRENNER, G., DA COSTA, J.L., DEVEZEAUX, D. and NIEDERDRENK, P. "Simulation of Viscous Flows past Hyperboloid Configurations with Flare," AIAA 29th Aerospace and Sciences Meeting, Reno, January 7-10, 1991.

15. BARILLOT, P., BOUSQUET, J.M., BRENNER, G., DA COSTA, J.L. and DEVEZEAUX, D. "Analysis Tools of ONERA and DLR for Aerothermodynamics of Reentry Vehicles," First European Symposium on Aerothermodynamics for Space Vehicles," Noordwijk, The Netherlands, May 28-30, 1991.

16. DA COSTA, J.L. "Hypersonic Inviscid Flow Field Simulations around Reentry Vehicles with Flap Deflection," First European Symposium on Aerothermodynamics for Space Vehicles," Noordwijk, The Netherlands, May 28-30, 1991.

17. DA COSTA, J.L., HACHEMIN and GUILLEN, Ph. "A Multidomain Euler Code Applied to Hypersonic Flows," AIAA 22nd Fluid Dynamics, Plasmadynamics and Laser Conference, Honolulu, Hawaii, June 24-26, 1991.

18. GAZAIX, M. "Three Dimensional Inviscid Real Gas Flow over a Double Ellipsoid," Workshop on Hypersonic Flows for Reentry Problems," Part II, Antibes, April 1991.

19. LE TOULLEC, L. and GUILLEN, Ph. "Hypersonic Delta Wing Flow Calculations Using a Multidomain MUSCL Euler Solver," Workshop on Hypersonic Flows for Reentry Problems," Part II, Antibes, April 1991.

20. DORMIEUX, M., GUILLEN, Ph. and ABGRALL, R. "Numerical Simulation of Transverse Jet Flows by a Non Reactive Two Species Multidomain Euler Solver," AIAA-Paper 1990-0126, 1990.

21. THEPOT, R. and DUVEAU, P. "Prediction Methods for Supersonic Inlets," 10th ISABE, Nottingham, U.K., 1-6 Sept. 1991.

22. BORREL, M., MONTAGNE, J.L., DIET, J., GUILLEN, Ph. and LORDON, J. "Upwind Scheme for Computing Supersonic Flows Around a Tactical Missile," La Recherche Aerospatiale 1988-2, (French and English versions), 1988.

23. GUILLEN, Ph. and LORDON, J. "Numerical Simulation of Separated Subsonic Flows Around Tactical Missile Bodies," AGARD Conference on "Validation of Computational Fluid Dynamics" AGARD CP-437, Paper 22, 1988.

24. RANOUX, G., LORDON, J. and DIET, J. "Geometrie et Maillage de Configurations Complexes Pour des Calculs Aerodynamiques," AGARD Conference on "Applications of Mesh Generation to Complex 3D Configurations," AGARD CP-464, Paper 5, 1989.

25. LACAU, R.G., and ROBERT, M. "Some Trends in Missile Aerodynamics," AGARD Conference on " Missile Aerodynamics," AGARD CP- 493, Paper 1, 1990.

26. SCHÖNE, J. and BIDAULT, J. "Calculations of Inviscid Flow Around a Reentry Configuration at Supersonic Speed," AIAA-Paper 91-0391, 1991.

27. MORTCHELEVICZ, G.D., "Solution of Euler Equations with Unstructured Mesh: 3-D Case," *La Recherche Aerospatiale*, Vol. 1991-6, pp. 17-25, French and English editions.

28. MORTCHELEVICZ, G.D. and SENS, A.S. "Solution of 3D Euler Equations with Unstructured Meshes for Aeroelasticity Problems," International Forum on Aeroelasticity and Structural Dynamics 1991, Aachen, June 1991.

29. LAVAL, P. and BILLET, G. "Recent Developments in Finite Difference Methods for the Computation of Transient Flows," in: *Numerical Methods for Transient and Coupled Problems*, John Wiley and Sons Ltd, Chapter 5, 1987.

30. DUTOYA, D. and ERRERA, M.P., "A Formal Decomposition of the Jacobian Matrix of the Euler Equations - Application to Upwind Numerical Schemes," *La Recherche Aerospatiale*, (English Edition), Vol. 1992-1, Jan./Febr. 1992.

31. KHALFALLAH, K., LACOMBE, G. and LERAT, A. " Inviscid and Viscous Flow Calculation over a Double Ellipse with and Implicit Centered Method," Proceedings of Workshop on Hypersonic Flows for Reentry Problems. Desideri, J.A., Glowinski, R. and Periaux, J. Eds. Springer Verlag, Vol. 1, 1991.

32. DUBOIS, F. and MICHAUX, O. "Solution of the Euler Equations around a Double Ellipsoidal Shape Using Unstructured Meshes and Including Real Gas Effects," Proceedings of Workshop on Hypersonic Flows for Reentry Problems. Desideri, J.A., Glowinski, R. and Periaux, J. Eds. Springer Verlag, Vol 2, 1992, pp. 358-373.

33. GASSIN, Ch. "Computation of Aerodynamic Coefficients on HERMES-ARIANE5 Configuration," Proceedings of the 1rst Symposium on Aerothermodynamics for Space Vehicles, ESTEC, Noordwijk, The Nederlands, May 28-30, 1991.

34. BREDIF, M., CHATTOT, J.J., KOECK, Ch. and WERLE, P. "Simulation d'un Systeme de Deviation de Jets a l'Aide des Equations d'Euler," AGARD CP-412 on "Applications of CFD on Aerodynamics," Aix-en-Provence, France, Paper 13, April 1986.

35. BOREL, C. and ROUX, F.X. "Implicit Multidomain Numerical Method for Unsteady Compressible Inviscid Fluid Flows around 3-D Projectiles," Proceedings of Parallel CFD' 91, (eds. K.G.Rensch et al.), Elsevier Science Publishers, 1992.

36. FEZOUI, L., STOUFFLET, B., PERIAUX, J. and DERVIEUX, A. "Implicit High-Order Finite Element Schemes for the Euler Equations," 4th Symposium on Numerical Methods in Engineering, Atlanta, USA, March 1986, Computational Mechanics Publ., Southampton UK.

37. STOUFFLET, B., PERIAUX, J., FEZOUI, L., and DERVIEUX, A. "3-D Hypersonic Euler Numerical Simulation around Space Vehicles using Adapted Finite Elements,", AIAA-Paper 87-0560, 1987.

38. BILLEY, V., DERVIEUX, A., FEZOUI, L., PERIAUX, J., SELMIN, V. and STOUFFLET, B., "Recent Improvements in Galerkin and Upwind Euler Solvers and Application to 3-D Transonic Flow in Aircraft Design," 8eme Colloque International sur les Methodes de Calcul Scientifique et Technique, Versailles Dec. 1987, Computer Methods in Applied Mechanics and Engineering, 1988.

39. ROSTAND, P. and STOUFFLET, B. "TVD Schemes to Compute Compressible Viscous Flows on Unstructured Meshes," Notes on Numerical Fluid Mechnics, Vol 24, pp. 510-520, Vieweg, Braunschweig, 1989.

40. FEZOUI, L. and STOUFFLET, B., "A Class of Implicit Upwind Schemes for Euler Simulations with Unstructured Meshes," *Journal of Computational Physics*, Vol. 84, 1989, pp. 174-206.

41. LECLERCQ, M.P., PERIAUX, J. and STOUFFLET, B. "Multigrid Methods with Unstructured Meshes," Proceedings of 7th International Conference on Finite Elements in Flow Problems, Huntsville, Alabama, 1989.

42. ARMINJON, P., DERVIEUX, A., FEZOUI, L., STEVE, H. and STOUFFLET, B. "Non oscillatory schemes for Multidimensional Euler Calculations with Unstructured Grids," in: Nonlinear Equations - Theory," Numerical Methods and Applications, Ballmann & Jeltsch Eds., pp. 1-10, Notes on Numerical Fluid Mechanics, 24, Vieweg, Braunschweig, 1989.

43. FEZOUI, L. and DERVIEUX, A. "Finite Element Non-Oscillatory Schemes for Compressible Flows," Symposium on Computational Mathematics and Applications, Pavie, Italy, October 1989.

44. GLINSKY, N., FEZOUI, L., CICCOLI, M.C. and DESIDERI, J.A. "Non Equilibrium Hypersonic Flow Computations by Implicit Second-Order Upwind Finite Elements," 8th GAMM Conf. on Numerical Methods in Fluid Dynamics, Notes on Numerical Fluid Mechnics, Vol. 29, Pieter Wesseling Ed., Vieweg 1990.

45. LARROUTUROU, B. and FEZOUI, L. "On the Equations of Multi-Component Perfect or Real Gas Inviscid Flow," in Nonlinear Hyperbolic Problems, (eds. Carasso, Charrier, Hanouzet and Joly), Lecture Notes in Mathematics, Springer Verlag, 1989.

46. LECLERC, M.P., MANTEL, B., PERIAUX, J., PERRIER, P. and STOUFFLET, B. "On Recent 3D Euler Computations around a Complete Aircraft using Adaptive Unstructured Mesh Refinements," Proceedings of Second World Congress on Computational Mechanics, Stuttgart, August 1990.

47. KASBARIAN, C., LECLERCQ, M.P., RAVACHOL, M. and STOUFFLET, B. "Improvements of Upwind Formulations on Unstructured Meshes," Proceedings of 4th International Conference on Hyperbolic Problems, Taormina, Italy, April 1992.

48. HIRSCH, Ch., LACOR, C., DENER, C. and VUCINIC, D. "An Integrated CFD System for 3D Turbomachinery Applications," AGARD CP-510, February 1992.

49. HIRSCH, Ch., RIZZI, A., LACOR, C., ELIASSON, P., LINDBLAD, J. and HÄUSER, J. "A Multiblock/Multigrid Code to Simulate Complex 3D Navier-Stokes Flows on Structured Meshes," Proceedings 4th Int. Symposium on CFD, UC Davis, Ca, USA, Sept. 1991.

50. LACOR, C., ALAVILLI, P., HIRSCH, Ch., ELIASSON, P., LINDBLAD, J. and RIZZI, A. "Hypersonic Navier-Stokes Computations About Complex Configurations," Proceedings 1rst European CFD Conf., Brussels, Belgium, Sept. 1992.

51. STRUYS, R., VANKEIRSBILCK, P. and DECONINCK, H., "An Adative Grid Polygonal Finite Volume Method for the Compressible Flow Equations" AIAA-Paper 89-1959-CP, pp. 303-311, 1989.

52. VANKEIRSBILCK, P. and DECONINCK, H., " Higher Order Upwind Finite Volume Schemes with ENO Properties for Unstructured Grids" in 'Unstructured Grid Methods for Advection Dominated Flows', AGARD R-787, Paper 7, 1992.

53. STRUYS, R., DEPALMA, P., ROE, P. L., and POWELL, K. G. "Progress on Multidimensional Upwind Euler Solvers for Unstructured Grids," AIAA Paper 91-1550, 1991.

54. DECONINCK, H., BOURGOIS, G., PAILLERE, H. and ROE, P.L., "Multidimensional Upwind Methods for Unstructured Grids," AGARD R-787, Paper 4, 1992.

55. MANNA, M. and DECONINCK, H. "Computation of 3-D Inviscid Hypersonic Flow around Sharp and Blunt Delta Wings," Proceedings of Workshop on Hypersonic Flows for Reentry Problems. Desideri, J.A., Glowinski, R. and Periaux, J. Eds. Springer Verlag, Vol. 2, 1991.

56. MANNA, M. "A 3D High Resolution Upwind Finite Volume Solver," Von Karman Institute TN180, 1992.

57. MANNA, M., DECONINCK, H., MA, E. and LI, C. P. "A Comparison of High Resolution Upwind Solvers on 3-D Inviscid Hypersonic Flow," AGARD-CP-514, 1992.

58. MENSINK, C., "A 2D Multiblock Computational Method for Viscous and Inviscid Compressible Flow, with Applications to Cascade Flows," VKI Technical Report, April 1992.

59. FORMAGGIA, L. "An Unstructured Mesh Generation Algorithm for 3D Aeronautical Configurations," Proceedings of the 3rd International Conference on Numerical Grid Generation in CFD and Related Fields, (eds. Arcilla A.S. et al.), Elsevier Science Publisher, 1991.

60. HETTENA, E., FORMAGGIA, L., ANTONELLINI, A. and SELMIN, V. "Validation of an Unstructured Euler Code for Simulation of Transonic Flows Past a Complete Aircraft," in Proceedings of the 13th ICNMFD, Rome, 1992, Lecture Notes in Physics, Vol 414, Springer-Verlag, Berlin Heidelberg 1993.

61. SELMIN, V., HETTENA, E. and FORMAGGIA, L. "An Unstructured Node Centered Scheme for the Simulation of 3D Inviscid Flow," Proceedings of the 1rst European CFD Conference, ECCOMAS, Brussels, 1992.

62. BORSI, FORMAGGIA, L., HETTENA, E., SANTILLAN, S., SELMIN, V. and TARDITI, S. "Vortical Flow Simulation Using Structured and Unstructured Grids," AGARD-CP-494, Paper 3, 1991.

63. SANTILLAN, S. and SELMIN, V. "Transonic Vortical Flow Predicted with a Structured Multiblock Euler Solver," Proceedings of the 1rst European CFD Conference, ECCOMAS, Brussels, 1992.

64. COIRO, D., AMATO, M. and de MATTEIS, P., "Numerical Prediction of Transonic Viscous Flows Around Aerofoils Through an Euler/Boundary Layer Interaction Method," *The Aeronautical Journal*, Vol. 96, No. 954, April 1992.

65. COIRO, D., de MATTEIS, P. and AMATO, M., "Wake Effects on the Prediction of Transonic Flows Around Airfoils," *Journal of Aircraft*, April 1992.

66. BORELLI, S. and PANDOLFI, M., "A Contribution to the Prediction of Hypersonic Non-Equilibrium Flows," Workshop on Hypersonic Flows for Reentry Problems, Antibes, Jan. 1990. (eds. Desideri, J. A., Glowinski, R. and Periaux, J.), Springer Verlag, Vol. 1, 1991.

67. BORELLI, S. and SCHIANO, P., "A Non-Equilibrium Hypersonic Flow Solver on Massively Parallel Computer," Parallel CFD '91 Conference, Stuttgart, June 1991.

68. BORELLI, S. MATRONE, P. and SCHIANO, P., "A Multiblock Hypersonic Flow Solver for Massively Parallel Computer," Parallel CFD '92 Conference, New Brunswick, N.J., USA, May 1992.

69. BORELLI, S. and SCHETTINO, "Influence of Chemical Modelling on Hypersonic Flow," 1rst European Symposium on Aerothermodynamics for Space Vehicles, ESTEC, Noordwijk, The Netherlands, May 1991.

70. BORELLI, S. RUSSO, G. and SCHETTINO, "A Contribution to the Analysis of High Enthalpy Nozzle Flows," Workshop on Hypersonic Flows for Reentry Problems Part II, Antibes, April 1991, (eds. Abgrall, R., Desideri, J. A., Glowinski, R., Mallet, M., Periaux, J.), Hypersonic Flows for Reentry Problems, Vol. III, Springer-Verlag, 1992.

71. FAVINI, B., "Assessment and Improvement of Fast Euler Solver," 8th GAMM Conf. on Numerical Methods in Fluid Mechanics, Delft, NLR, (ed. Wesseling, P.), Notes on Numerical Fluid Mechanics, Vol. 23, pp. 119-127, 1989.

72. SABETTA, F., FAVINI, B., MORETTI, G., ONOFRI, M. and VALORANI, M. "Reactive and Inert Inviscid Flow Solution by Quasi-Linear Formulations and Shock-Fitting," Proceedings of Workshop on Hypersonic Flows for Reentry Problems. (eds. Desideri, J. A., Glowinski, R. and Periaux, J.), Springer Verlag, Vol. 1, 1991.

73. VALORANI, M., ONOFRI, M., FAVINI, B. and SABETTA, F. "Non-Equilibrium Hypersonic Inviscid Steady Flows," *AIAA Journal*, Vol. 30, 1992, pp. 86-93.

74. SABETTA, F., FAVINI, B. and ONOFRI, M., "Influence of Chemical Modelling on the Solution of Hypersonic Shock Layers," 1rst European Symposium on Aerothermodynamics for Space Vehicles, ESTEC, Noordwijk, The Netherlands, May 1991.

75. FAVINI, B. and ZANNETTI, L., "On Conservation Properties and Non-Conservative Forms of Euler Solvers," 10th ICNMFD, Beijing, Lecture Notes in Physics, Vol. 264, Springer Verlag, pp. 270-275, 1986.

76. FAVINI, B. and ZANNETTI, L., "A Difference Scheme for Weak Solutions of Gasdynamic Equations," AIAA-Paper 87-0537, 1987.

77. Di MASCIO, A. and FAVINI, B., "A Two-Step Godunov-type Scheme for Multidimensional Compressible Flows," Proceedings 9th GAMM Conference on Numerical Methods in Fluid Mechanics, Lausanne 1991.

78. THOMADAKIS M. P. and TSANGARIS S. "Improved Artificial Dissipation Schemes for the Euler Equations," *International Journal for Numerical Methods in Fluids*, Vol. 14, 1992, pp. 1391-1405.

79. THOMADAKIS M. P. and TSANGARIS S. "Transonic Unsteady Inviscid and Viscous Flow Simulation Around 2D Moving Bodies," AIAA-Paper 92-2704, 1992.

80. THOMADAKIS M. P. and TSANGARIS S. "On the Predicition of Transonic Unsteady Flows Using Second Order Time Accuracy," Computational Fluid Dynamics '92, Vol. 2, (eds. Hirsch Ch. et al), Elsevier Science Publisher pp. 711-718, 1992.

81. TSANGARIS S. and DRIKAKIS D. "Multigrid Scheme for the Compressible Euler Equations," *ZAMM*, Vol. 70/6, 1990, T664-666.

82. DRIKAKIS D. and TSANGARIS S. "On the Accuracy and Efficiency of CFD Methods in Real Gas Hypersonics," *International Journal for Numerical Methods in Fluids*, Vol. 16, 1993, pp. 759-775.

83. DRIKAKIS D. and TSANGARIS S. "An Implicit Characteristic-Flux-Averaging Method for the Euler Equations for Real Gases," *International Journal for Numerical Methods in Fluids*, Vol. 12, 1991, pp. 711-726.

84. DRIKAKIS D. and TSANGARIS S. "Local Solution Acceleration Method for the Euler and Navier-Stokes Equations," *AIAA-Journal*, Vol. 30, 1992, pp. 340-348.

85. DRIKAKIS D. and TSANGARIS S. "Real Gas Effects for Compressible Nozzle Flows," *Journal of Fluids Engineering, Transaction of ASME*, Vol. 115, 1993, pp. 115-120.

86. SOTIROPOULOS F., ALKALAI K., GIOANNAKOGLOU K. and PAPAILIOU K.D. "Application of a Fractional Step Algorithm to the Solution of Inviscid Turbomachinery Flow Problems," Proc. of the 109th WAM of ASME, Chicago, USA, Nov. 1988, also in: "Flows in Non-Rotating Turbomachinary Components," FED - Vol. 69, eds. Rohatgi V.S., Hamed A. and Kim J.H., 1988.

87. OKTAY E., AKMANDOR, I. S. and ÜÇER A. S. "Three-Dimensional Solution of Internal Flows Using a Cell-Vertex Finite-Volume Method," AGARD CP-510, 1992.

3.3 CODES FROM GERMANY AND SWITZERLAND

3.3.1 Overview

In the following the essential Euler codes from Germany and Switzerland are presented. Most codes being developed in industry follow a structured approach whereby either a flexible monoblock or a general multiblock strategy is applied to accomplish the treatment of complex geometries including full configurations of aircraft, missile and space vehicles. To achieve a reasonable level of productivity with respect to flow computations most industries have interfaced their codes to a dedicated grid generation system providing all necessary block-interface informations for the solver. Another observation which can be made is the fact that almost all general purpose Euler-codes can also be used for viscous simulation by solving the Navier-Stokes equations.

Research and development of new approaches concerning discretization schemes and solution techniques take place mainly at universities and research establishments. From there a broad variety of codes are reported which mostly are limited in their ability to treat geometrically complex problems or physical models.

To characterize the major individual Euler-codes from industry and research institutions by their underlying basic schemes and the intended range of applications corresponding short descriptions are provided below. The presentation is ordered according to the institutions where codes were developed.

MELINA:
At the DASA Airbus Division company essentially the codes developed at the DLR-Institute for Design Aerodynamics in Braunschweig are used and implemented as the code baseline.

However further development and refinement for production purposes takes place within the company.

The finite-volume cell-vertex code MELINA is the result of these efforts. The code is multiblock-structured and is applied mostly to transonic problems for transport aircraft design.[1-3] MELINA is based on an explicit 5-stage Runge-Kutta time stepping scheme and a cell-centered discretization combined with the blended artificial dissipation operator as proposed by Jameson et al.[4] Convergence acceleration is provided by a full approximation (FAS) multigrid method whereas increased spatial accuracy can be achieved by a block-oriented local grid refinement capability. Recently the code was extended to a Navier-Stokes solver.

Apart from routine tasks as the flow simulation around wings and wing/body configurations emphasis is on support for laminar wing design, integration of propulsion systems and flap track fairing design.

Grid generation is tailored to design oriented tasks and is provided by the INGRID system, an in-house development.

IKARUS :
At the DASA Regional Turboprop Division, which is formed by the Dornier Luftfahrt company, the general purpose code IKARUS is in continuous development since 1984. The code development was initiated from and based essentially on the work of Jameson and Schmidt.[4-6] The 3-D version of that basic approach was Jameson's FLO-57 from which an early multiblock version was derived by Jameson and Leicher[7] forming the basis for Dornier's IKARUS code.

The structured multiblock code IKARUS is based on a cell-centered, finite-volume discretization for solving the steady and unsteady, compressible 3-D Euler and Navier-Stokes equations in integral form, thus providing numerical solutions to inviscid and viscous flow problems in almost arbitrary geometries. The treatment of complex geometries is facilitated by an advanced multiblock technique allowing a general segmentation of block faces leading to a high flexibility concerning the application, the connection and the built-up of any types of grid block topologies. Explicit and implicit integration schemes are available based on the same cell-centered spatial discretization assuring identical steady state results.

Explicit integration is performed by a linear multistage Runge-Kutta-type time-stepping scheme. Convergence acceleration is provided by several techniques like local time-stepping, implicit residual averaging, enthapy forcing and a FAS multigrid scheme, offering V- and W-cycle options.

Implicit steady state solutions are also enabled by an approximate Newton-method operating on the steady state equations. The corresponding relaxation method is called the LU-SSOR scheme which can be used together with the multigrid option to enhance convergence.

From early 3-D flow simulations past wings and wing/body combinations[8,9] the code was also applied to inviscid flow problems around inlet[10,11] and propfan[12] configurations as well as vortical flows past delta wings.[13-16] Moreover IKARUS was successfully applied for great number of flow problems ranging from in the low subsonic (incompressible) up to the hypersonic flow range[17] which includes also simulations based on the equilibrium real gas assumption.[18,19]

The flow solver IKARUS is logically complemented by an interactive grid generator called DOGRID which allows a fast and easy generation of complex blockstructured meshes.[20,21]

EUFLEX, INFLEX, ROTFLEX:
Essentially based on the work of Eberle at DASA Military Aircraft Division (formerly MBB/UF) a 3-D Euler code, called EUFLEX, is continuously developed since 1984. In the meantime the code has achieved a high level of theoretical sophistication. For these reasons the code is applied with great success in practical aerodynamic project work as a general purpose design tool not only in the military aircraft division but also at the corresponding missile, helicopter and propulsion divisions of DASA.

EUFLEX is based on a monoblock, finite-volume method for solving the integral form of the conservation laws for inviscid flow. A "Godunov-type" differencing approach is followed, enforcing the efficient solution of the Riemann problem at each cell face for definition of interface states.

Because of the very special approach some essential aspects of the method are described in more detail. The basic theory[22-24] starts the development by considering the Riemann problem in the non-conservative, differential form of the Euler equations resulting in an exact, iteration-free solution to the Riemann problem. However the basic approach has the disadvantage that into the corresponding solution of the primitive solution vector at the interface also entropy values from right or left states enter. In practical computations this fact lead to non-negligible entropy errors. To overcome that problem an successful attempt is made to find a solution of the Riemann-problem in terms of the conservative variables. In the development of the so-called "homogeneous" Riemann-solver properties and assumptions of acoustic wave theory are exploited. Essentially the isentropic transport of entropy along streamlines is used to define appropriate interface pressure values. Using the homogeneous property of the Euler fluxes then allows the transformation to the corresponding Riemann solution in terms of conservative variables.

Into the relations defining the conservative variables at the interface as a solution of a locally one-dimensional Riemann-problem, values for the conservative variables associated with the individual eigenvalues have to be determined. This is performed by a third order interpolation between left and right states. A symmetric weighted average of corresponding eigenvalues between left and right states are taken. By that the

scheme is kept fully differentiable, preventing any glitches in the solution whenever eigenvalues are changing sign. This process is called the "characteristic flux averaging" scheme. Higher order accuracy is achieved by the MUSCL-approach in concert with an improved van Albada-flux limiter. Corresponding non-oscillatory extrapolation formulas are given and tested up to fifth order.[24,25]

In the original version of the code time integration to steady state was performed by an explicit scheme supported by the local time stepping technique. Acceptable convergence rates, however, could only be realized with point relaxation schemes.[26] Some problems are reported concerning possible singular implicit block operator matrices for high aspect-ratio cells.[25] As a remedy a local transformation to non-conservative [illegible]des is proposed, which avoids this ill-conditioning prob-

[illegible]cheme outlined so far was applied to a number of real life [illegible]ases, with the simulation of the inviscid flow around a complete fighter aircraft as an exceptional highlight at the time published.[26] A special effort was made to increase code robustness for hypersonic applications.[24,27-30] This was due to the observation that the method produced pre-shock spikes during the shock movement process for very strong shocks resulting in possible negative pressures. This is overcome by a generalized flux formulation in which the Riemann-flux vectors and the Beam/Warming split flux vectors are combined such, that at strong shocks the Riemann-fluxes are switched off smoothly and the more stable Beam/Warming fluxes are activated. The success of the proposed approach is demonstrated by results of intake[28] and nozzle[29,30] flow computations. A comprehensive description of the overall approach is given in the textbook of Eberle, Hirschel and Rizzi.[31]

The treatment of complex configurations is enabled by a monoblock technique in which for each cell the desired boundary conditions are specified. Also available is multiblock version of the EUFLEX-code. A Navier-Stokes version, derived from the EUFLEX-code is developed as NSFLEX[32].

Code development was also extended to the time-accurate treatment of unsteady flow problems. These efforts led to the unsteady version, called INFLEX. 3-D applications were made to oscillating, but rigid wings in pitching motion.[32-34] For time integration an implicit unfactored relaxation scheme is employed, which allows a dramatic shortening of computation time as compared to the explicit scheme. Careful studies are conducted concerning the effect of different convergence criteria controlling the iteration process during one time step on the solution.[34]

At the EUROCOPTER Deutschland company, a joint company of AEROSPATIALE and DASA in the helicopter business EUFLEX is modified and adjusted to rotorcraft problems.[35,36] The resulting ROTFLEX code includes an unsteady and time-accurate computation capability.[36]

DAINV-SPACE, DAINV-SPLIT, DAVIS-VOL:

In the DASA Space Infrastructures Division the corresponding aerodynamic department is working for more than a decade on inviscid flow problems essentially related to supersonic external and nozzle type flows. In recent years strong emphasis is on hypersonic reentry and cruise problems. Hypersonic flow problems are emerging from the European and national projects focussing on the reentry vehicle HERMES and the two-stage space transportation system SÄNGER. Continuous efforts have led to the development of a series of Euler codes, called DAINV-SPACE, DAINV-SPLIT and DAVIS-VOL, which will be characterized subsequently.

DAINV-SPACE[37-40] is being developed since many years and belongs to the class of supersonic space-marching codes. The numerical approach, originally based on a central-difference scheme, has been developed to a split-matrix finite difference method solving the quasiconservative and steady-state form of the inviscid conservation laws. The outer shock wave is fitted to the outer computational region (shock fitting approach) whereas shock waves embedded inside the shock layer are captured. The code is able to deal with perfect and equilibrium gas conditions.[38] An upwind discretization approach is used relying on sign-splitting of the associated flux Jacobians. Some efforts are made to accomplish the eigenvalue decomposition with an appropriate set of eigenvectors. Originally space integration was performed with an explicit Runge-Kutta scheme. For stability reasons each space integration step is now considered as the steady state solution of a time-dependent problem. Some emphasis is put on the formulation of the wall boundary condition, which is carried out in the frame of characteristic compatibility relations. Results are presented for supersonic flows past a cone at incidence as well as for a delta wing and a generic configuration of a hypersonic space transportation system.[39,40]

The DAINV-SPLIT[41-48] code is based on a finite-difference method for solving the time-dependent and quasi-conservative form of the 2-D and 3-D Euler equations. A diagonalisation of the flux Jacobians is performed to allow the proper implementation of a second order upwind scheme according to the sign of the associated eigenvalues.[41-43] An explicit multi-stage Runge-Kutta time integration scheme is used in concert with local time stepping to achieve steady state solutions. Solid wall boundary conditions are based on characteristic compatibility relations. The code has the capability for shock and shear layer fitting and can treat flows under perfect, equilibrium as well as non-equilibrium gas conditions. First applications of the method are reported to nozzle flow problems[44] and flow simulations past the forebody as well as complete configurations of a reeentry vehicle under equilibrium real gas conditions.[45,46]

The DAINV-SPLIT code is also extended to account for non-equilibrium chemistry.[47,48] Chemical non-equilibrium conditions add to the conservation equations for inviscid flow additional equations accounting for the mass conservation of the reacting gas species. For low pressures, representative for reentry situations, the specfic enthalpy of the various species can be considered as only depending on temperature, whereas the mixture of reacting gases is assumed to behave according to Dalton's law. So formally the non-equilibrium code option has been developed along similar concepts already proved for ideal or equilibrium real gas applications. The well-known Park's 5 species/17-reactions model without ionization is used and integrated by an explicit multistage time-stepping scheme. However, if in the chemistry model the various reaction time scales are smaller by orders of magnitude compared to the characteristic time scales of the locally frozen mixture then the chemistry is determining locally the time step size for the whole system of equations. In this situation the code switches to a point-implicit scheme for time integration of the source term appearing in the species mass conservation equations.

Interestingly DAINV-SPLIT has an option to operate also with an enthalpy correction technique improving the convergence rate to steady state as well as code robustness in transient solutions stages. This is due to the fact that a constant freestream total enthalpy is enforced in the whole flowfield. An additional option is called "pseudo space marching". By that acronym the possibility is understood to march over the flowfield in streamwise direction with a stack of several grid planes. This technique provides a steady state solution for a specific grid plane in the middle of the stack. Assuming supersonic flow and a proper implementation of a pure upwind scheme the final steady state flowfield solution should be identical to a global solution approach where at each time step a sweep over the whole solution domain has to be carried out. Results are presented for non-equilibrium flows past a cylinder and a sphere which are compared to available experimental data.[47]

For shock fitting a general formulation of the Rankine-Hugoniot relations was developed valid for ideal, equilibrium and nonequilibrium real gas in a co-moving coordinte system.[48] A procedure is outlined to reduce the general shock relations resulting in an efficient numerical scheme. Results are presented for the flow past a complete HERMES configuration under non-equilibrium conditions at Mach 25.[48]

DAVIS-VOL[49] is a numerical method for solution of the 2-D and 3-D strong conservative form of the integral conservation laws. The approach follows the finite-volume time stepping concept. The numerical fluxes are evaluated according to a symmetric TVD scheme essentially proposed by Yee with the definition of the eigenvector matrices at the cell faces according to Roe. An implicit point or line Gauss-Seidel relaxation scheme is adopted for time integration. In the associated implicit operator the fluxes are approximated to first-order accuracy using the Roe-averaged, sign-splitted flux Jacobians.

To accelerate convergence to steady state the local time stepping technique is applied together with a so-called "switch-off" scheme, which controls the iteration process such that only those cells are updated where a change of the solution vector beyond some prescribed criteria can be anticipated. As an option for supersonic flow problems the DAVIS-VOL code is also able to operate in the "pseudo-marching" mode (described previously), leading to a considerable gain in computational efficiency for appropppriate problems. This is demonstrated for the flow past a HERMES forebody under ideal and equilibrium real gas, assumptions. Using the pseudo-marching technique a gain of a factor 10 is claimed compared to the global solution approach. Applications are also presented for the 3-D interacting flow of a two-fin model configuration.

The natural extension of the scheme is also presented to handle viscous flow problems by solution of the Navier-Stokes equations.[40]

MTU-EULER:
In the DASA Jet Engine Division (MTU Munich) quasi 3-D (in S1 blade-to-blade and S2 hub-to-tip planes) as well as full 3-D Euler codes are under development since 1985.

The quasi 3-D approach as described by Happel et al.[50] is based on the 2-D approximation of the conservation laws along stream surfaces taking into account the streamwise mean radius of curvature of the surface as well as the stream tube thickness. For flow simulation past rotor blades the conservation laws are formulated in a rotating frame of reference The numerical scheme is based on a finite-volume cell-vertex discretization in which the convective fluxes are approximated to first order in space. An explicit first order time-stepping scheme is used to integrate the unsteady equations to steady state. Convergence acceleration is provided by a local time-stepping and by a multigrid technique. A so-called "damping surface technique" is applied to provide a post-correction of the results to enhance the spatial accuracy. The method is applied to transonic flow problems past turbine stator and rotor blades as well as to compressor cascades.

A full 3-D code extension is presented by Happel and Stuberts[51] aiming on flow simulations past complete blade rows. Interesting comparisions between experimental data and results from 3-D Euler and Navier-Stokes computations in the cascade rig of an inlet guide vane of a low pressure turbine are reported by Niehuis et al.[52]

CATS, CEVCATS:
In the German Aerospace Research Establishment (DLR) systematic development work on Euler codes as tools for aerodynamic design is performed mainly at the DLR-Institute for Design Aerodynamics in Braunschweig.

Euler code development has started there in 1983/84 and was a major focus point for theoretical work over the years. The basic approach was the finite-volume concept outlined by Jameson et al.[8] which led to a 3-D, block-structured, cell-centered and central-differenced code, called CATS, described by Radespiel and Kroll.[53,55] Convergence acceleration of the baseline Runge-Kutta-type linear multistage time-stepping scheme for steady state problems is achieved by local time-stepping, enthalpy damping and implicit residual averaging techniques. Flexibility of the blockstructured code has been increased subsequently by introduction of a mesh embedding technique allowing the change of mesh density by a factor of 2 for portions of a mesh block. Corresponding improvement of the method concerning efficiency in terms of computer time for specified accuracy was demonstrated by Radespiel[54] for a transport aircraft wing/body combination.

A cell-vertex variant of the CATS-code, called CEVCATS, was introduced by Rossow[56] and Rossow et al.[57] CEVCATS has been developed further to a general purpose fluid simulation package for the whole speed range, also handling viscous flows as a Navier-Stokes solver. Application examples are found in Kroll et al.[58] where a variety of aerospace configurations are analyzed operating in the transonic as well as supersonic flow regime. Transport aircraft type wing/body as well as generic canard-delta wing configurations were treated successfully in subsonic and transonic flow, whereas a generic fighter type forebody, a waverider type delta wing and an early configuration of the European HERMES reentry vehicle were studied at supersonic flow conditions.

A FAS multigrid technique for acceleration of convergence to steady state is implemented into CEVCATS based on the work of Radespiel and Swanson[59]. A discussion on experiences with multigrid techniques applied to high supersonic flow fields (Ma<10) can be found in Kroll et al.[60] A method for proper implementation of the baseline multigrid techniques in the multiblock framework of CEVCATS is discussed by Atkins.[61]

In recent years CEVCATS has found widespread application in aerodynamic design problems. The work of Schöne et al.[62,63] reflect this effort whereas CEVCATS and an implementation of the symmetric TVD discretization scheme of Yee is compared[62] with respect of accuracy and shock resolution issues for a flow around blunted biconic at Mach 6. A comparison of two Euler codes, namely CEVCATS and FLU3C, is presented in Schöne et al.[63] FLU3C is developed at ONERA and is based on a flux-vector splitting technique according to van Leer. Extensive and detailed comparisons have been made for supersonic flow computations around a vertical tail alternate reentry vehicle configuration to the baseline HERMES concept.

Another topic of continuous research are transport aircraft design problems. Due to increasing importance of optimal airframe integration of present and future high bypass ratio jet engines systematic experimental and theoretical studies were conducted to investigate the position as well as the influence of thrusted and unthrusted jet engine operation on aerodynamic wing characteristics. Hoheisel et al.[64] used a turbo-powered simulator for low speed experiments at Mach 0.17 for basic investigations. A corresponding theoretical study taking into account a future ultra high bypass jet engine with considerable higher nacelle diameter as compared to existing engines was analyzed by Rossow[65]. The inviscid simulation of an isolated thrusted high-bypass jet engine was considered by Rudnik.[66] Whereas in such inviscid simulations the influence of pylon/nacelle combinations on aerodynamic wing performance due to pure displacement effects can be studied quite successfully, interference effects due to viscous jet stream mixing processes are neglected. An integrated attempt of several computational tools including CEVCATS for possible design of a laminar flow nacelle is reported by Radespiel et al.[67]

Research on inviscid flow analysis for slender wings by Euler methods is a major topic since 1986. Kumar and Das[68] studied subsonic and transonic flow around a cropped delta wing configuration defined for purpose of Euler code validation as part of the US/European vortex flow experiment project. The CATS code was applied to investigate sharp and round leading edges up to high angles of attack producing vortex flow breakdown. Extensions of that work to a coupled canard/delta configuration using CEVCATS is presented subsequently by Scherr and Das.[69] In an attempt to clarify the role of dissipation in Euler solutions of different codes for vortical flows over delta wings Longo[70] made systematic comparisons concerning the effect of mesh density and level of artificial dissipation in the CATS/CEVCATS codes, respectively the level of truncation errors in the EUFLEX code on global and local data. Detailed compari-

sions between experimental and theoretical results for close-coupled canard-wing and double-delta wing are presented by Longo and Das[71]. It was demonstrated that primary vortices shed from leading edges are captured quite well. However cross-sectional total pressure surveys show distinct quantitative differences between simulation and experiments which may be attributed to viscous effects and missing resolution of vortex sheets.

Thorough discussions on the most suitable numerical formulations related to rotating frame of reference for flow simulations past propellers and hovering rotors can be found in Kroll.[72] The corresponding rotorcraft version of the CATS code was applied to steady state problems around two-blade propellers.[73]

An optimum design method where the CEVCATS code was coupled to an optimization algorithm has been developed by Schöne.[74] The method was applied to the design of conical and general 3-D wings at supersonic speeds.

Recent efforts at DLR are related to the enhancement of spatial accuracy by the use of high resolution schemes[75,76] and grid refinement strategies.[77] Systematic work is underway concerning the improvement of the various algorithmic elements of the multigrid technique.[78]

Beside development efforts directed to the CEVCATS code some efforts aiming on unstructured methods have to be mentioned.[79]

Euler-Codes from Research Institutions and Universities

DLR-Institute for Theoretical Fluid Mechanics:
At the DLR-Institute for Theoretical Fluid Mechanics in Göttingen no continuous efforts are made for development of a general purpose Euler-code. However basic work on high resolution discretization schemes by Müller et al.[80] and corresponding applications to hypersonic waverider problems[81] are reported.

EPFL-Ecole Polytechnique Federal de Lausanne:
At the EPFL-Institut des Machines Hydrauliques et de Mechanique des Fluides in Lausanne cooperative efforts together with CERFACS (France) are made towards the development of a 3-D general purpose multiblock finite-volume Euler code[82] including equilibrium and non-equilibrium real gas capability.[83-85] The cell-centered discretization approach is based on Jameson's aritificial dissipation operator[4] and for time integration an explicit linear multistage time-stepping scheme is applied. Another direction of Euler-code application concerns incompressible flow simulation problems in water turbines.[86-88]

Institut für Strahlantriebe und Turboarbeitsmaschinen, RWTH-Aachen:
At the RWTH-Institut für Strahlantriebe und Turboarbeitsmaschinen efforts are made for development of a general purpose finite-volume code aiming at steady and unsteady turbomachinery flow applications.[89-91] Besides structured also unstructured codes are under development.[92]

Institut für Aero- und Gasdynamik, Universität Stuttgart:
At the Institut für Aero- und Gasdynamik main efforts are directed to development of a finite-volume Euler code aiming at rotorcraft flow applications.[93-95] Basis for development is the EUFLEX code from the DASA Military Aircraft Division.

Institut für Raumfahrtsysteme, Universität Stuttgart :
At the Institut für Raumfahrtsysteme main efforts are directed to development of a 3-D finite-volume Euler code aiming at turbomachinery flow applications.[96,97]

Institute for Computer Applications, Universität Stuttgart:
At the Institute for Computer Applications serious efforts are underway for development of a general purpose finite-element Euler/Navier-Stokes code including equilibrium and non-equilibrium real gas capabilities.[98,99] Focus point are reentry flow applications related to the European reentry vehicle project HERMES.

3.3.2 Presentation of Individual Codes

COMPANY / INSTITUTION: DASA - Airbus Division / Bremen - Germany

CODE	DISCRET. TECHNIQUE	GRIDS	SPACE/TIME DISCRET.	CONVERGENCE ACCELERATION	SPECIAL FEATURES	REF.	POC
MELINA	3-D, finite-volume, cell-vertex	multiblock-structured, blockwise mesh refinement	central differencing, Jameson's second and fourth difference dissipation operator, explicit Runge-Kutta scheme	local time stepping, implicit residual smoothing, enthalpy damping, multigrid	Euler/Navier-Stokes solver, actuator b.c., shock capturing	1-3	1

COMPANY / INSTITUTION: DASA - Regioprop Division / Dornier Luftfahrt / Friedrichshafen - Germany

CODE	DISCRET. TECHNIQUE	GRIDS	SPACE/TIME DISCRET.	CONVERGENCE ACCELERATION	SPECIAL FEATURES	REF.	POC
IKARUS	3-D, finite-volume, cell-centered	multiblock-structured, blockwise mesh refinement, arbitrary block face segmentation	central differencing, Jameson's blended second and fourth difference dissipation operator, Runge-Kutta explicit or LU-SSOR implicit scheme	local time stepping, implicit residual smoothing, enthalpy damping, multigrid	Euler/Navier-Stokes solver, actuator b.c., shock capturing, equilibrium real gas option	7-21	2

COMPANY / INSTITUTION: DASA - Military Aircraft Division / Ottobrunn - Germany							
CODE	DISCRET. TECHNIQUE	GRIDS	SPACE/TIME DISCRET.	CONVERGENCE ACCELERATION	SPECIAL FEATURES	REF.	POC
EUFLEX	3-D, finite-volume, cell-centered	monoblock-structured	upwind (combined Steger flux vector and Godunov -type) differencing, point Gauss-Seidel implicit scheme	local time stepping	Navier-Stokes version: "NSFLEX", shock capturing, equilibrium real gas option	22-31	3
INFLEX	3-D, finite-volume, cell-centered	monoblock-structured	upwind (combined Steger flux vector and Godunov -type) differencing, point Gauss-Seidel implicit scheme, 1st order Euler backward time operator		time accurate version of EUFLEX	32-34	

COMPANY / INSTITUTION: EUROCOPTER Deutschland / Ottobrunn - Germany							
CODE	DISCRET. TECHNIQUE	GRIDS	SPACE/TIME DISCRET.	CONVERGENCE ACCELERATION	SPECIAL FEATURES	REF.	POC
ROTFLEX	3-D, finite-volume, cell-centered	monoblock-structured	upwind (combined Steger flux vector and Godunov -type) differencing, point Gauss-Seidel implicit scheme	local time stepping,	rotorcraft version of EUFLEX, also time-accurate version available	35,36	4

COMPANY / INSTITUTION: DASA - Space Infrastrucures / Ottobrunn - Germany							
CODE	DISCRET. TECHNIQUE	GRIDS	SPACE/TIME DISCRET.	CONVERGENCE ACCELERATION	SPECIAL FEATURES	REF.	POC
DAINV-SPACE	2-D & 3-D, finite-difference	single block, structured	upwind flux-vector splitting scheme, Runge-Kutta explicit space/time integration scheme		space marching method, shock fitting, equilibrium gas option	37-40	5
DAINV-SPLIT	2-D & 3-D, finite-difference	multiblock-structured	upwind (flux-vector splitting) differencing, Runge-Kutta explicit time stepping scheme / point Gauss-Seidel implicit for species source terms	local time stepping	shock fitting, quasi-conservative Euler formulation, perfect gas, equilibrium and non-equilibrium real gas option	41-48	
DAVIS-VOL	2-D & 3-D, finite-volume, cell-centered	multiblock-structured	central (symmetric TVD) differencing, implicit Euler backward time stepping, symmetric Gauss-Seidel relaxation	local time stepping	Euler/Navier-Stokes solver, equilibrium real gas option	49	

COMPANY / INSTITUTION: DASA - Jet Engine Division (MTU - Munich) - Germany							
CODE	DISCRET. TECHNIQUE	GRIDS	SPACE/TIME DISCRET.	CONVERGENCE ACCELERATION	SPECIAL FEATURES	REF.	POC
MTU-EULER	2-D & 3-D, finite-volume, cell-vertex	multiblock-structured	central differencing with second difference dissipation operator, explicit first-order Euler time stepping scheme	local time stepping, multigrid	shock capturing, post-correction of results by "damping surface" procedure	50-52	6

COMPANY / INSTITUTION: DLR - Institute for Design Aerodynamics / Brunswick - Germany							
CODE	DISCRET. TECHNIQUE	GRIDS	SPACE/TIME DISCRET.	CONVERGENCE ACCELERATION	SPECIAL FEATURES	REF.	POC
CATS	3-D, finite-volume, cell-centered	multiblock-structured, blockwise mesh refinement	central differencing, Jameson's blended second and fourth difference dissipation operator, explicit Runge-Kutta time stepping scheme	local time stepping, implicit residual smoothing, enthalpy damping, multigrid	shock capturing	53-55	7

COMPANY / INSTITUTION: DLR - Institute for Design Aerodynamics / Brunswick - Germany							
CODE	DISCRET. TECHNIQUE	GRIDS	SPACE/TIME DISCRET.	CONVERGENCE ACCELERATION	SPECIAL FEATURES	REF.	POC
CEVCATS	3-D, finite-volume, cell-vertex	multiblock-structured, blockwise mesh refinement	central (Jameson's blended second and fourth difference dissipation operator, symmetric TVD scheme) and upwind differencing, explicit Runge-Kutta time stepping scheme	local time stepping, implicit residual smoothing, enthalpy damping, multigrid	Euler/Navier-Stokes solver, actuator bc., shock capturing, equilibrium real gas, general config. capability	56-75	7

COMPANY / INSTITUTION: DLR - Institute for Theoretical Fluid Mechanics / Göttingen - Germany							
CODE	DISCRET. TECHNIQUE	GRIDS	SPACE/TIME DISCRET.	CONVERGENCE ACCELERATION	SPECIAL FEATURES	REF.	POC
(ITS-Euler)	3-D, finite-difference	single block, structured	upwind (Harten-Yee) TVD scheme, 3-factor AF implicit Beam-Warming scheme	local time stepping	shock capturing	80,81	8

COMPANY / INSTITUTION: EPFL - Institute / Lausanne - Switzerland							
CODE	DISCRET. TECHNIQUE	GRIDS	SPACE/TIME DISCRET.	CONVERGENCE ACCELERATION	SPECIAL FEATURES	REF.	POC
(EPFL-Euler)	3-D, finite-volume, cell-centered	multiblock-structured	central differencing, Jameson's blended second and fourth difference dissipation operator, explicit Runge-Kutta time stepping scheme	local time stepping, residual averaging	shock capturing, equilibrium and non-equilibrium real gas option	82-88	9

3.3.3 Points of Contact

Point of Contact (POC) No. 1:

Code(s): MELINA
Name: S. Rill
Dept.: EF10
Tel.: (+49) 421-538-4499
Mailing Address: Deutsche Aerospace Airbus GmbH
D-28183 Bremen
Germany
Tel.-Company: (+49) 421-538-01
Fax-Company: (+49) 421-538-3320
References: 1, 2

Point of Contact (POC) No. 2:

Code(s): IKARUS
Name: H. Rieger / S. Leicher
Dept.: LREV3
Tel.: (+49) 7545-84203 /-84819
Mailing Address: Dornier Luftfahrt GmbH
D-88039 Friedrichshafen
Germany
Tel.-Company: (+49) 7545-80
Fax-Company: (+49) 7545-84411
References: 7 - 21

Point of Contact (POC) No. 3:

Code(s): EUFLEX, INFLEX
Name: A. Eberle
Dept.: LME211
Tel.: (+49) 89-607-24912
Mailing Address: Deutsche Aerospace AG
Military Aircraft Division
D-81663 München
Germany
Tel.-Company: (+49) 89-607-0
Fax-Company: (+49) 89-607-26481
References: 22 - 34

Point of Contact (POC) No. 4:

Code(s): ROTFLEX
Name: H. Stahl-Cucinelli
Dept.: D/EE41
Tel.: (+49) 89-607-23681
Mailing Address: Eurocopter Deutschland GmbH
D-81663 München
Germany
Tel.-Company: (+49) 89-607-0
Fax-Company: (+49) 89-607-26888
References: 35, 36

Point of Contact (POC) No. 5:

Code(s): DAINV-SPACE, DAINV-SPLIT, DAVIS-VOL
Name: C. Weiland
Dept.: RIT73
Tel.: (+49) 89-607-28473
Mailing Address: Deutsche Aerospace AG
Space Infrastructures
D-81663 München
Germany
Tel.-Company: (+49) 89-607-0
Fax-Company: (+49) 89-607-26481

References: 37 - 49

Point of Contact (POC) No. 6:

Code(s): MTU-EULER
Name: H.-W. Happel
Dept.: EWTS
Tel.: (+49) 89-1489-2535
Mailing Address: MTU Motoren- und Turbinen-Union
München GmbH
D-80991 München
Germany
Tel.-Company: (+49) 89-14 89-0
Fax-Company: (+49) 89-150 2621

References: 50 - 52

Point of Contact (POC) No. 7:

Code(s): CATS, CEVCATS
Name: N. Kroll / R. Radespiel
Dept.: Numerische Aerodynamik
Tel.: (+49) 531-295-2440 / -2488
Mailing Address: Deutsche Forschungsanstalt für
Luft- und Raumfahrt e.V. (DLR)
Institut für Entwurfsaerodynamik
Lilienthalplatz 7
D-38108 Braunschweig
Germany
Tel.-Company: (+49) 531-295-0
Fax-Company: (+49) 531-295-2320

References: 53 - 75

Point of Contact (POC) No. 8:

Code(s): ITS-EULER
Name: W. Kordulla
Dept.:
Tel.: (+49) 551 709-2274 or -2275
Mailing Address: Deutsche Forschungsanstalt für
Luft- und Raumfahrt e.V. (DLR)
Inst. für Theoretische Strömungsmechanik
SM-SM
Bunsenstraße 10
D-37073 Göttingen
Germany
Tel.-Company: (+49) 551 709-1
Fax-Company: (+49) 551 709-2446

References: 80, 81

Point of Contact (POC) No. 9:

Code(s): EPFL-EULER
Name: C. Bergmann
Dept.:
Tel.: (+21) 693-3503 or -3504
Mailing Address: Swiss Federal Institue of Technology
Institut de Machines Hydraulique
et de Méchanique des Fluides
Bâtiment DME
EPFL-Ecublens
CH-1015 Lausanne
Switzerland
Company: (+21) 693-1111
Fax-Company: (+21) 693-3646 or -2525

References: 82 - 88

Point of Contact (POC) No. 10:

Code(s): RWTH-EULER
Name: H.E. Gallus
Dept.:
Tel.: (+49) 241 80-5500
Mailing Address: Rheinisch-Westfälische
Technische Hochschule Aachen (RWTH)
Institut für
Strahlantriebe und Turbomaschinen
Templergraben 55
D-52062 Aachen
Germany
Tel.-Company: (+49) 241 80-5504
Fax-Company: (+49) 241 8888-229

References: 89 - 92

Point of Contact (POC) No. 11:

Code(s): IAGS-EULER
Name: S. Wagner
Dept.:
Tel.: (+49) 711-685-3580
Mailing Address: Universität Stuttgart
Institut für Aero- und Gasdynamik
Pfaffenwaldring 21
D-70569 Stuttgart
Germany
Tel.-Company: (+49) 711-685-1
Fax-Company: (+49) 711-685-3438 or -3500

References: 93 - 95

Point of Contact (POC) No. 12:

Code(s): IRS-EULER
Name: H.-H. Frühauf
Dept.:
Tel.: (+49) 711 685-2382
Mailing Address: Universität Stuttgart
Institut für Raumfahrtsysteme
Pfaffenwaldring 31
D-70569 Stuttgart
Germany
Tel.-Company: (+49) 711-685-1
Fax-Company: (+49) 711-685-3596 or -3500

References: 96 - 97

Point of Contact (POC) No. 13:.

Code(s): ICA-EULER
Name: J. Argyris
Dept.:
Tel.: (+49) 711-685-3594
Mailing Address: Universität Stuttgart
Institut für Computer-Anwendungen
Pfaffenwaldring 27
D-70569 Stuttgart
Germany
Tel.-Company: (+49) 711-685-1
Fax-Company: (+49) 711-685-3669 or -3500

References: 98, 99

3.3.4 References

1. RILL, S. and BECKER, K. "Simulation of Transonic Inviscid Flow Over a Twin Jet Transport Aircraft," AIAA-Paper 91-0025, 1991

2. RILL, S. "A High-Resolution, Implicit Scheme for the Solution of the Euler Equations," DGLR-Paper 89-167 (1989), pp. 377-386.

3. RILL, S. "MELINA - A Multi-Block, Multi-Grid 3-D Euler Code with Local Sub-Block Technique for Local Mesh Refinement," ICAS-Paper 92-4.3.R, ICAS Conf. Beijing, Sept. 1992.

4. JAMESON, A., SCHMIDT, W. and TURKEL, E. "Numerical Solutions of the Euler Equations by Finite Volume Methods Using Runge-Kutta Time Stepping Schemes," AIAA-Paper 81-1259, 1981.

5. SCHMIDT, W., JAMESON, A. and WHITFIELD, D. "Finite Volume Solutions to the Euler Equations," *Journal of Aircraft*, Vol. 20, 1983, pp. 127-133.

6. SCHMIDT, W. and JAMESON, A. "Euler Solvers as an Analysis Tool for Aircraft Aerodynamics," in: Advances in Computational Transonics, (ed. Habashi, W. G.), Pineridge Press, Swansea, U.K., 1985, Paper 13, pp. 371-404.

7. JAMESON, A., LEICHER, S. and DAWSON, J. "Remarks on the Development of a Multiblock Three-Dimensional Euler Code for Out of Core and Multiprocesor Calculations," in: Progress in Scientific Computing Vol. 6 (1985) pp. 53-66, (Proceedings of U.S.-Israel Workshop on "Progress and Supercomputing in Computational Fluid Dynamics," Jerusalem, 1984).

8. LEICHER, S. "Analysis of Transonic and Supersonic Flows Around Wing-Body Combinations," Proc. of the 14th Congress of the International Council of the Aeronautical Sciences, ICAS-84-1.2.2, 1984.

9. LEICHER, S. "Numerical Simulation of Internal and External Inviscid and Viscous 3D Flow Fields," AGARD-CP-412, Paper 14, 1986.

10. GRASHOF, J. "Numerical Investigation of Three-Dimensional Transonic Flow Through Air Intakes Disturbed by a Missile Plume," AIAA-Paper 83-1854, 1983

11. BUERS, H., LEICHER, S. and MACKRODT, P.A. "Numerical and Experimental Investigation of Engine Inlet Flow with the Dornier EM2 Supersonic Inlet Model," AGARD-CP 437, Paper 32, 1988.

12. LEICHER, S., LÜCKING, P. and [illegible]LITZ, P. "Calculation of Flow Along a Cowl of a Sh[illegible] Propfan Using a 3D Euler-code," Proc. of the 16th C[illegible] of the International Council of the Aeronautical Sciences, [illegible]-88-5.1.4, 1988.

13. HITZEL, S. "Wing Vortex-Flows Up Into Vortex Breakdown - A Numerical Simulation," AIAA-Paper 88-2518, 1988.

14. HITZEL, S. M. and WAGNER, B. "Computations Relating to the Vortex Flow Experiment," Proc. of the 3rd Symposium Transonicum, Göttingen-Germany, May 26, 1988.

15. WAGNER, B., HITZEL, S. M., SCHMATZ, M.A., SCHWARZ, W., HILGENSTOCK, A. and SCHERR, S. "Status of CFD Validation on the Vortex Flow Experiment," AGARD-CP 437, Paper 10, 1988.

16. HITZEL, S. M. "Low and High Speed High Angle of Attack Flow Around a Delta Wing by an Euler Simulation," Proc. of the Int. Conf. on The Prediction and Expoitation of Separated Flow, Royal Aeronautical Society, London, April 18-20, 1989, Paper 24

17. RIEGER, H. and JAMESON, A. "Solution of Steady Three-Dimensional Compressible Euler and Navier-Stokes Equations by an Implicit LU Scheme," AIAA-Paper 88-0619, 1988.

18. RIEGER, H., STOCK, H. and WAGNER, B. "Computational Aerothermodynamic Methods For Industrial Applications To Reentry And Hypersonic Cruise Problems," ESA-SP-318, 1991, pp. 421-437.

19. DORTMANN, K. "Viscous and Inviscid Hypersonic Flow About a Double Ellipsoid," Proc. of the Workshop on Hypersonic Flows for Reentry Problems, (eds. Desideri J.-A. et al.), Vol. 2, pp. 616-634, Springer Verlag, 1991.

20. SEIBERT, W. "A Graphic-Interactive Program-System to Generate Composite Grids for General Configurations," Numerical Grid Generation in Computational Fluid Dynamics, Pineridge Press, 1988, pp. 517-528.

21. SEIBERT, W., FRITZ, W. and LEICHER, S. "On the Way to an Integrated Mesh Generation System for Industrial Applications," AGARD-CP 464, Paper 14, 1989.

22. EBERLE, A. "MBB-EUFLEX - A New Flux Extrapolation Scheme Solving the Euler Equations for Arbitrary 3-D Geometry and Speed," MBB/LKE122/S/PUB/140, 1984.

23. EBERLE, A. "3D Euler Calculations Using Characteristic Flux Extrapolation," AIAA-Paper 85-0119, 1985.

24. EBERLE, A. "Characteristic Flux Averaging Approach to the Solution of Euler's Equations," VKI Lecture Series 1987-04, March 1987.

25. EBERLE, A., SCHÄFER, O. and SCHMATZ, M. A. "High order Solutions of the Euler Equations Using Characteristic Flux Averaging," 12th Congress of the International Council of the Aeronautical Sciences, ICAS-86-1.3.1, 1986.

26. EBERLE, A. and MISEGADES, K. "Euler Solution for a Complete Fighter Aircraft at Sub- and Supersonic Speed," Applications of Computational Fluid Dynamics in Aeronautics, AGARD-CP 412, Paper 17, 1986.

27. EBERLE, A., SCHMATZ, M. A. and BISSINGER, N. C. "Generalized Flux Vectors for Hypersonic Shock Capturing," AIAA-Paper 90-0390, 1990.

28. BISSINGER, N. C. and EBERLE, A. "CFD Contributions During Hypersonic Airplane Intake Design," AGARD-CP 510, Paper 26, 1991.

29. EBERLE, A. "Enhanced Numerical Inviscid and Viscous Fluxes for Cell Centered Finite Volume Schemes," Proc. of the 4th Int. Symposium on Computational Fluid Dynamics, Sept. 9-12, 1991,University of Davis, Davis, Vol.1, pp. 296-301

30. EBERLE, A. "Characteristic Fluxvectors for Implicit High Speed Flow Solutions," Advances in Computational Fluid Dynamics, (eds. Habashi W. G. and Hafez M. M.), Pineridge Press Ltd, Swansea, 1992.

31. EBERLE, A., RIZZI, A. and HIRSCHEL, E.H. "Numerical Solutions of the Euler Equations For Steady Flow Problems," Notes on Numerical Fluid Mechanics, Vieweg Verlag, Braunschweig-Wiesbaden, Vol. 34, 1991.

32. SCHMATZ, M.A., BRENNEIS, A. and EBERLE, A. "Verification of an Implicit Relaxation Method for Steady and Unsteady Viscous and Inviscid Flow Problems," AGARD-CP-437, Vol. 1, Paper 15, 1988.

33. BRENNEIS, A. and EBERLE, A. "Unsteady Transonic Flows past Airfoils and Wings Using a Fast Implicit Godunov Type Euler Solver," Proc. of the 16th Congress of the International Council of the Aeronautical Sciences, Jerusalem, 1988, ICAS-Paper 88.6.3.1

34. BRENNEIS, A. and EBERLE, A. "Application of an Implicit Relaxation Method Solving the Euler Equations for Real Unsteady Problems," *Journal of Fluids Engineering*, Vol.112, 1990, pp. 510-520.

35. STAHL, H. "Application of a 3D Euler Code to Transonic Blade Tip Flow,"Proc. of the 12th European Rotorcraft Forum, Garmisch-Partenkirchen- Germany, Sept. 22-25, 1986, Paper 29.

36. STAHL-CUCINELLI, H. "Application of 3D Euler Code To Rotor Blade Tips," Proc. of the 15th European Rotorcraft Forum, Amsterdam, Sept. 12.-15, 1989, Paper 15.

37. WEILAND, C. "Calculation of Three -Dimensional Stationary Supersonic Flow Fields by Applying the 'Progonka' Process to a Conservative Formulation of the Governing Equations," *Journal of Computational Physics*, Vol. 29, 1978, pp. 173-198.

38. WEILAND, C., PFITZNER, M. and HARTMANN, G. "Euler Solvers For Hypersonic Aerothermodynamic Problems," Proc. of the 7th GAMM-Conference on Numerical Methods in Fluid Mechanics, in: Notes on Numerical Fluid Mechanics, Vol. 20, 1988, pp. 426-433.

39. WEILAND, C. "A Split-Matrix Runge-Kutta Type Space Marching Procedure," *Journal of Computational Physics*, Vol. 102, 1992, pp. 319-335.

40. MENNE, S. and WEILAND, C. "Calculation of Three-Dimensional Viscous and Inviscid Hypersonic Flows Using Split-Matrix Marching Methods," AIAA-Paper 90-3070, 1990.

41. WEILAND, C. "A Split-Matrix Method for the Integration of the Quasi-Conservative Euler-Equations," Proc. of the 6th GAMM-Conf. on Numerical Methods in Fluid Mechanics, in: Notes on Numerical Fluid Dynamics, Vol. 13, 1986, pp. 383-390.

42. WEILAND, C. and PFITZNER, M. "3-D and 2-D Solutions of the Quasi-Conservative Euler Equations," Proc. of the 10th Int. Conf. on Numerical Methods in Fluid Dynamics, in: Lecture Notes in Physics, Vol. 264, 1986, pp. 654-659.

43. PFITZNER, M. and WEILAND, C. "3-D Euler Solution For Hypersonic Mach Number," AGARD-CP. 428, Paper 22, 1987.

44. RIEDELBAUCH, S. and WEILAND, C. "Inviscid Laval-Nozzle Flowfield Calculation," *Journal of Spacecraft and Rockets*, Vol. 25, 1988, pp. 88-90.

45. PFITZNER, M. "Runge-Kutta Split-Matrix Method For the Simulation of Real Gas Hypersonic Flow," Proc. of the Second Int. Conf. on Nonlinear Hyperbolic Problems, in: Notes on Numerical Fluid Mechanics, Vol. 24, 1989, pp. 489-498.

46. HARTMANN, G. and WEILAND, C. "Strömungsfeldberechnung um den HERMES Raumgleiter während der Wiedereintrittsphase," DLR-Jahrbuch 1991, Vol. 1, 1991, Paper 91-223, pp. 255-264 (in german)

47. PFITZNER, M. "Simulations of Inviscid Equilibrium and Non-Equilibrium Hypersonic Flows," Proceedings of the 12th Int. Conf. on Numerical Methods in Fluid Dynamics, in: Lecture Notes in Physics, Vol. 371, pp. 432-436, Springer Verlag, 1990.

48. PFITZNER, M. "A 3-D Non-Equilibrium Shock-Fitting Algorithm Using Effective Rankine-Hugoniot Relations," AIAA-Paper 91-1467, 1991.

49. PFITZNER, M., SCHRÖDER, W., MENNE, S. and WEILAND, C. "Three-Dimensional Simulations of Hypersonic Flows," Proc. of the Int. Conf. on Hypersonic Aerodynamics, University of Manchester, Sept. 4.-6., 1989.

50. HAPPEL, H.-W., DIETRICHS, H. J. and LEHMANN, K. "Computation of Transonic 2D Cascade Flow and Comparison with Experiments," AGARD-CP 401, Paper 19, 1986.

51. HAPPEL, H.-W. and STUBERT, B. "Computation of Transonic 3D Cascade Flow and Comparison with Experiments," AGARD-CP 437, Paper 31, 1988.

52. NIEHUIS, R., LÜCKING, P. and STUBERT, B. "Experimental and Numerical Study on Basic Phenomena of Secondary Flows in Turbines," AGARD-CP 468/469, Paper 5, 1989.

53. RADESPIEL, R. and KROLL, N. "Progress in the Development of an Efficient Finite Volume Code for the Three-Dimensional Euler Equations," DFVLR-FB-85-31, 1985

54. RADESPIEL, R. "Efficient Solution of Three-Dimensional Euler Equations Using Embedded Grids," Proc. of 15th ICAS Congress, Paper No. 86-1.3.3, London, UK, September 1986.

55. KROLL, N. and JAIN, R. K. "Solution of Two-Dimensional Euler Equations - Experience with a Finite Volume Code," DFVLR-FB 87-41, 1987.

56. ROSSOW, C.-C. "Comparison of Cell Centered and Cell Vertex Finite Volume Schemes," Notes on Numerical Fluid Mechanics, Vol. 20, Vieweg-Verlag, 1988.

57. ROSSOW, C.-C., KROLL, N. RADESPIEL, R. and SCHERR, S. " Investigation of the Accuracy of Finite Volume Methods for 2- and 3-Dimensional Flows," AGARD-CP-437, Vol. 2, Paper 14, 1988

58. KROLL, N., ROSSOW, C.-C., SCHERR, S., SCHONE, J. and WICHMANN, G. "Analysis of 3D-Aerospace Configurations Using the Euler Equations," AIAA-Paper 89-0268, 1989.

59. RADESPIEL, R. and SWANSON, R. C. "An Investigation of Cell Centered and Cell Vertex Multigrid Schemes for the Navier-Stokes Equations," AIAA-Paper 89-0548, 1989

60. KROLL, N., RADESPIEL, R. and ROSSOW, C.-C. "Experience with Explicit Time-Stepping Schemes for Supersonic Flow Fields," Notes on Numerical Fluid Mechanics, Vol. 29, Vieweg Verlag, 1990.

61. ATKINS, H. "A Multiple-Block Multigrid Method for the Solution of the Three-Dimensional Euler and Navier-Stokes Equations," DLR-FB 90-45, 1990

62. SCHÖNE, J., KROLL, N., ROSSOW, C.-C., LI, H. and SONAR, T. "A Central Finite Volume TVD Scheme for the Calculation of Supersonic and Hypersonic Flow Fields Around Complex Configurations," AIAA-Paper 89-1975, 1989.

63. SCHONE, J. and BIDAULT, J. "Calculations of Inviscid Flow Around a Reentry Configuration at Supersonic Speed," AIAA-Paper 91-0391, 1991.

64. HOHEISEL, H., KIOCK, R., ROSSOW, C.-C, RONZHEIMER, A., BAUMERT, W. and CAPDEVILA, H. "Aspects of Theoretical and Experimental Investigations on Airframe/Engine Integration Problems." ICAS Congress, Stockholm, Paper 90-2.7.3., 1990.

65. ROSSOW, C.-C., GODDARD, J.-L., HOHEISEL, H. and SCHMIDT, V. "Investigation of Propulsion Integration Interference Effects on a Transport Aircraft Configuration," AIAA-Paper 92-3097, 1992.

66. RUDNIK, R. "Erweiterung eines dreidimensionalen Euler-Verfahrens zur Berechnung des Strömungsfeldes um Nebenstromtriebwerke mit Fan- und Kernstrahl," DLR-FB 91-13, 1991 (in german).

67. RADESPIEL, R., HORSTMANN, K.-H. and REDEKER, G. "Feasibility Study on the Design of a Laminar Flow Nacelle," AIAA-Paper 89-0640, 1989.

68. KUMAR, A. and DAS, A. "Numerical Solultion of Flow Fields Around Delta Wings Using Euler Equation Method," Proc. of Symposium on International Vortex Flow Experiment on Euler Code Validation, pp. 175-216, Stockholm, 1986.

69. SCHERR, S. and DAS, A. "Basic Analysis of the Flow Fields of Slender Delta Wings Using the Euler Equations," ICAS Congress Jerusalem, 1988, Vol. II, pp. 1428-1436

70. LONGO, J. M. A. "The Role of the Numerical Dissipation on the Computational Euler Equations Solutions for Vortical Flows," AIAA-Paper 89-2232, 1989.

71. LONGO, J .M. A. and DAS, A. "Numerical Simulation of Vortical Flows over Close-Coupled Canard-Wing Configuration," AIAA-Paper 90-3003, 1990

72. KROLL, N. "Berechnung von Strömungsfeldern um Propeller und Rotoren im Schwebeflug durch die Lösung der Euler-Gleichungen," DLR-FB 89-37, 1989 (in german).

73. KROLL, N. "Computations of the Flow Fields of Propellers and Hovering Rotors Using Euler Equations," Proceedings of 12th European Rotorcraft Forum, Paper 28, 1986.

74. SCHÖNE, J. "Design of Supersonic Wing Using an Optimization Strategy Coupled with a Solution Scheme for the Euler-Equations," AIAA-Paper 90-3060, 1990.

75. ROSSOW, C.-C. "Flux Balance Splitting - A New Approach for a Cell Vertex Upwind Scheme," Proceedings of 12th International Conference on Numerical Methods in Fluid Dynamics, Oxford, 1990.

76. KROLL, N., GAITONDE, D., and AFTOSMIS, M. "A Systematic "Comparative Study of Several High Resolution Schemes for Complex Problems in High Speed Flows," AIAA-Paper 91-0636, 1991.

77. AFTOSMIS, M. and KROLL, N. "A Quadrilateral Based Second-Order TVD Method for Unstructured Adaptive Meshes," AIAA-Paper 91-0124, 1991.

78. BLAZEK, J., KROLL, N., RADESPIEL, R. and ROSSOW, C.-C. "Upwind Implicit Residual Smoothing Method For Multi-Stage Schemes," AIAA-Paper 91-1533, 1991.

79. DIMIER, A. and RONZHEIMER, A. "Solution of the Euler Equations on Unstructured and Adaptive Meshes, DLR-IB 129-90/44, 1990

80. MÜLLER, B. and YANG, J. Y. "Performance Comparison of Second-Order ENO and TVD Schemes for the Euler Equations of Gas Dynamics," 3rd Int. Conf. on Hyperbolic Problems, Uppsala, Sweden, June 11-15, 1990.

81. MÜLLER, B., NIEDERDRENK, P. and SOBIECZKY, H. "Simulation of Hypersonic Waverider Flow," 1st Int. Hypersonic Waverider Symposium, University of Maryland, Oct. 17-19, 1990.

82. VOS, J. B. and BERGMANN, C. M. "The Development of a Multi-Block Flow Solver for Parallel Computers," Report IMHEF T-90-7, 1990.

83. VOS, J. B. and BERGMANN, C. M. "Chemical Equilibrium and Non-Equilibrium Inviscid Flow Simulations Using an Explicit Scheme," Report IMHEF T-90-10, 1990.

84. VOS, J. B. and BERGMANN, C. M. "Inviscid Hypersonic Flow Simulations Using an Explicit Scheme," Report IMHEF T-90-23, 1990

85. VOS, J. B. BONOMI, E. and BERGMANN, C.M. "A 2D Euler Solver for Real Gas Flows using an Adaptive Grid," Notes on Fluid Mechanics Vol. 29, pp. 554-563, Vieweg Verlag, Braunschweig, 1990

86. ERIKSSON, L. E. , RIZZI, A. and THEREE, J.-P. "Numerical Solutions of the Steady Incompressible Euler Equations Applied to Water Turbines," AIAA-Paper 84-2145, 1984.

87. SAXER, A., FELICI, H., NEURY, C. and RYHMING I. L. "Euler Flows in Hydraulic Turbines and Ducts Related to Boundary Conditions Formulation," in: Proc. of the Seventh GAMM-Conference on Numerical Methods in Fluid Mechanics, M. Deville ed., Vol 20, Vieweg Verlag, 1988, pp. 343-354.

88. THIBAUD, F. , DROTZ, A., and SOTTAS, G. "Validation of an Euler Code for Hydraulic Turbines," AGARD Symp. Validation Comp. Fluid Dyn., AGARD-CP-437, Vol. 1, Paper 27, 1988.

89. KAU, H.-P. and GALLUS, H. E. "Numerical Investigation of Unsteady Flow in Oscillating Turbine and Compressor Cascades," AGARD-CP-468/469 Paper 2, 1989.

90. BENETSCHIK, H. and GALLUS, H. E. "Inviscid and Viscous Flows in Transonic and Supersonic Cascades Using an Implicit Upwind Relaxation Algorithm," AIAA-Paper 90-2128, 1990.

91. BENETSCHIK, H. and GALLUS, H. E. "Inviscid and Viscous Transonic Flows in Cascades Using an Implicit Upwind Algorithm," *Journal of Propulsion and Power*, 1991.

92. KOSCHEL, W., LÖTZERICH, M. and VORNBERGER, A. "Solution on Unstructured Grids for the Euler- and Navier-Stokes Equations," AGARD-CP-437, Paper 26, 1988.

93. KRÄMER, E., HERTEL, J. and WAGNER, S. "A Study of the Influence of a Helicopter Rotor Blade on the Following Blades Using Euler Equations," 14th European Rotorcraft Forum, Milano, Italy, Sept. 20-23, 1988.

94. HERTEL, J. and KRÄMER, E. and WAGNER, S. "Complete Euler-Solution for a Hovering Rotor and a Propeller in Forward Flight," 16th European Rotorcraft Forum, Glasgow, U.K., Sept. 18-21, 1990.

95. KRÄMER, E., HERTEL, J. and WAGNER, S. "Euler Procedure for Calculation of the Steady Rotor Flow with Emphasis on Wake Evolution," AIAA-Paper 90-3007, 1990.

96. LECHELER, S. and FRÜHAUF, H.-H. "An Accurate and Fast 3-D Euler-Solver for Turbomachinery Flow Calculation," 12th International Conference on Numerical Methods in Fluid Dynamics, Oxford, England, July 9-13, 1990.

97. LECHELER, S. and FRÜHAUF, H.-H. "A Fully Implicit 3-D Euler-Solver for Accurate and Fast Turbomachinery Flow Calculation," 9th GAMM Conf. on "Numerical Methods in Fluid Mechanics," Lausanne, Switzerland, Sept. 25-27, 1991.

98. ARGYRIS, J., DOLTSINIS, I. ST. and FRIZ, H. "Hermes Space Shuttle: Exploration of Reentry Aerodynamics," *Computational Methods in Applied Mechanics and Engineering*, Vol. 73, 1989, pp. 1-51.

99. ARGYRIS, J., DOLTSINIS, I. ST. and FRIZ, H. "Studies on Computational Reentry Aerodynamics," *Computational Methods in Applied Mechanics and Engineering*, Vol. 81, 1990, pp. 257-289.

3.4 CODES FROM GREAT BRITAIN, THE NETHERLANDS and SCANDINAVIA

3.4.1 Overview

The majority of solution techniques routinely used to solve the Euler equations are based upon the Lax-Wendroff (structured grids), Taylor-Galerkin (unstructured grids), and Jameson explicitly added artificial dissipation with Runge-Kutta time stepping methods (structured and unstructured grids). Structured grids are primarily used but recently there appears to be growing interest within industry for the use of unstructured grids with their inherent advantages for adaptation techniques and flexibility for complex geometries.

Major uses of the multiblock grid and flow procedure for complex three-dimensional geometries include the National Aerospace Laboratory (NLR), The Netherlands, with the code ESOLVB, British Aerospace (BAe - Commercial Aircraft) with the code EJ83, SAAB, Sweden, with the code MULTEUL and CFD norway, Norway, with the code ThreeFlow. For the solution techniques it is interesting to note that they all use the classic Jameson[1,2] procedure with the main features of cell-centered central differencing, explicit local time stepping, added artificial dissipation with combinations of second and fourth order differences.

In the **United Kingdom**, the British Aerospace code EJ83[3] has multigrid convergence acceleration and a viscous capability with options for viscous simulation with a coupled boundary layer based upon 2D strip theory or full 3D boundary layer theory. In addition to their multiblock code, British Aerospace have developed a suite of codes using the Jameson solution technique for particular applications. The code EJ61, which is applicable to two-dimensional aerofoils has a full adaptation capability with both point refinement and node movement, EJ50 and EJ53 are specifically used for intakes and forebody geometries, EJ63 and EJ65 for wing/body configurations. The latter two codes, in common with the multiblock code EJ83, have a viscous coupled capability.

The Defence Research Agency (DRA), Farnborough, UK have developed several different codes for the solution of the Euler equations. The technique used to simulate the flows over isolated wings is based on the work of Ni[4] and enhanced by Hall[5] and relies upon the Lax-Wendroff method with a multigrid capability. Their unsteady capability uses an implicit Beam and Warming[6] scheme with central differencing approximations and an artificial dissipation approach.

Oxford University, UK have developed Euler solvers for both two and three-dimensional geometries which utilise a two parameter Lax-Wendroff time stepping with node based multigrid and Jameson artificial viscosity terms.[7-9] The solvers are applicable to single block grids, in particular, C-H grids for isolated wings.

In isolation within the survey, Swansea University and Computational Dynamics Research, Swansea, UK have developed the codes BRITE3D, FLITE3D, TG and HYBRID each of which is based upon unstructured triangular and tetrahedral grids.[10-18]

The code BRITE3D and FLITE are both based upon the Jameson method[19,20] and have a face and side based data structure implementation, respectively. BRITE3D contains a tetrahedral grid generator based upon the Delaunay triangulation[10-12] and the FLITE3D has an advancing front generator. The FLITE3D solver also has a multigrid capability[18]. Both have grid adaptation capabilities based upon the methods of grid remeshing, point refinement and the use of sources. The code TG is based upon the Taylor-Galerkin formulation on unstructured grids.[13,14] All the codes BRITE3D, FLITE3D and TG are applicable to a wide range of aerodynamic configurations. The code HYBRID[18] has a cell-centered Runge-Kutta time stepping algorithm which is applicable to unstructured, structured and hybrid grids. These grids can be automatically generated and both the grid and flow are executed through a menu driven MOTIF user interface. The code also has a viscous capability with options of a Baldwin-Lomax and k-ε turbulence models and grid adaptation on any grid type can be implemented using point refinement, point derefinement, remeshing, source adaptation and point movement. It is primarily used in the research environment.

In **The Netherlands**, at the National Aerospace Laboratory NLR, the software system ENFLOW for the calculation of Euler flows around complex aerodynamic configurations is available. The system may also be used for the calculation of Navier-Stokes flows. In particular, ENFLOW is applied for the computational analysis of the aerodynamics of flows around transport aircrafts, including the aerodynamic effects of propellers and/or jet-engines. Based on early technical concepts[21-23] the system was implemented in cooperation with FOKKER of The Netherlands. ALENIA and CIRA, both of Italy, participated in the development of an early version of the system. ENFLOW as a CFD user environment is operational on a diversity of hardware and available in various releases for industry and research institutions.

To treat the aerodynamic flow simulation task efficiently with the ENFLOW system a decomposition into five subtasks has been accomplished. By such a design it became possible to isolate critical topics like geometry manipulation and grid generation from the other CFD work.[24,27,29]

The five subtasks are the geometric surface modelling and appropriate modification of aerodynamic surfaces, the decomposition of the flow domain into subdomains or blocks[21-24,27-28] by the software ENDOMO, the multiblock mesh generation[25,26,27,28] the flow calculation with the flow solver ENSOLV[29,30,32,35,37,38] and finally the graphical interactive data analysis and flow visualization.[40] The various code elements within the system are interfaced by standarized file formats.

The multiblock grids which are accepted by the flow solver ENSOLV should have the following characteristics.[21-29,37] Blocks are patched to each other, without gaps or overlaps. Block-faces are allowed to be sub-structured or segmented, which gives additional flexibility concerning the handling of complex boundary conditions. The code ENSOLV has also the capability for a block-by-block grid adaptation using point refinement.

ENSOLV is based on a cell-centered finite-volume discretization using explicit second and fourth difference dissipation operators for treatment of convective terms according to Jameson[1,2]. Integration to steady state is performed by various multi-stage Runge-Kutta schemes. Convergence speed is essentially accelerated by a multigrid procedure offering full or semi-coarsening options. As much as 14 different kinds of so-called external boundary conditions are accepted, including inlet, outlet and propeller boundary conditions, the latter being modelled as actuator disk. Second order accuracy is maintained also across discontinuous block interfaces by special block-coupling routines.

The ENFLOW system has been applied successfully to a manifold of inviscid flow problems. Reported are simulations past transport aircrafts[41] including propeller slipstream effects[30,31] and past delta wings.[33,34,36,39]

NLR is developing also an unstructured flow solver, called D2EUL[44]. It is based on a finite-volume cell vertex discretisation with flux difference upwinding[42] and second order accuracy extension according to the MUSCL scheme.[43] Grid generation based on triangles is automated and requires minimal user interaction. The code is highly vectorized and is planned to be extended to a 3D capability.

The Aerospace Faculty of Delft University of Technology, The Netherlands, has developed an Euler solver for three-dimensional geometries,[45] which is an extension of a 2D solver[46] from CWI, The Netherlands. This code[45] uses several

upwind-biased discretization techniques (van Leer, Osher and Roe) and an unfactored relaxation method within a multigrid solution procedure. The solver is applicable to single block structured grids.

In **Sweden**, SAAB relies essentially on their multiblock code MULTEUL[47] for the treatment of general three-dimensional complex configurations. The code is able to handle general flow simulations past complete missile, aircraft and aerospace vehicles. As indicated by the name, the MULTEUL code has built in a multigrid convergence acceleration technique. Mesh generation is based on a commercial system, called ICEM-MULCAD. It should be noted that MULTEUL has also a viscous option enabling Navier-Stokes simulations.

In addition to MULTEUL a general space-marching code, called GEMINI[48] is in productive use at SAAB for treatment of supersonic flow problems. According to a specific time/space integration technique the code circumvents the integration step size restrictions posed by stability reasons. Essentially the unknown steady state at the new space position is determined by solution of a time-dependent problem. Which means that the originally hyperbolic problem in space is converted to a problem being hyperbolic in time. This approach leads to improved algorithmic stability and makes the extension to parabolized Navier-Stokes solver straightforward. The grid is structured and is automatically generated plane-by-plane with an algebraic interpolation procedure at each section, as needed. Therefore, the code GEMINI is applied for inviscid and viscous flow simulations past general missiles and aircraft configurations with emphasis on aerodynamic flows around wings, rudders, inlets and diverters.

In **Norway** a key institution in the field of flow simulation is the company "CFD norway as". The company has developed a complete package consisting of grid generators, flow solvers and visualization tools for the treatment of 2-D and 3-D flows in complex geometries that are based on a structured multiblock approach.

The grid generation package uses algebraic and elliptic-type as well as combined techniques for generating blockstructured grids.[49] The flow solvers[50] are based on a cell-centered finite-volume discretization using the switched second and fourth order dissipation operators according to Jameson[1] for approximating the convective terms. Time integration is performed with a linear 3-stage Runge-Kutta scheme. The flow solvers include also options for solving either the thin-layer or the full Navier-Stokes equations with several zero-equation and two-equation eddy viscosity turbulence models. Options for operating the code in a rotating frame of reference as well as with real gas assumptions are available.

3.4.2 Presentation of Individual Codes

COMPANY / INSTITUTION: BRITISH AEROSPACE (BAe) - Commercial Aircraft Division / United Kingdom

CODE	DISCRET. TECHNIQUE	GRIDS	SPACE/TIME DISCRET.	CONVERGENCE ACCELERATION	SPECIAL FEATURES	REF.	POC
EJ61	2-D, finite-volume	structured	central differencing, Jameson-type and fourth difference dissipation operator, explicit multistage Runge-Kutta scheme	local time stepping	coupling to 2D b.l.	3	1
EJ50 EJ53 EJ63 EJ65	3-D, finite-volume		central differencing, Jameson-type and fourth difference dissipation operator, explicit multistage Runge-Kutta scheme	local time stepping	E5550,563 pure Euler, E563,E565 viscous coupling to 2D strip theory and full 3-D b.l.	3	
EJ83	3-D, finite-volume	multiblock-structured	central differencing, Jameson-type and fourth difference dissipation operator, explicit multistage Runge-Kutta scheme	local time stepping	viscous coupling to 2D strip theory and full 3-D b.l.	3	

COMPANY / INSTITUTION: : DEFENSE RESEARCH AGENCY (DRA) / Farnborough - United Kingdom

CODE	DISCRET. TECHNIQUE	GRIDS	SPACE/TIME DISCRET.	CONVERGENCE ACCELERATION	SPECIAL FEATURES	REF.	POC
ROE/LITTON	2-D, finite-volume	single grid, structured, O-type	Riemann solver, explicit multistage Runge-Kutta scheme	local time stepping			2
HALL-SALMOND-LITTON	3-D, finite-volume, cell-vertex	single grid, structured, C-H type	explicit Lax-Wendroff scheme	local time stepping, multigrid	isolated wing applications	4, 5	
EMJET	3-D, finite-volume cell-vertex	single grid, structured cartesian box	explicit Lax-Wendroff scheme	local time stepping, multigrid	b.c of jet orifice	4, 5	
UNSTEADY EULER	2-D/3-D finite-volume, cell-centered	structured	central differencing, Jameson-type second and fourth difference dissipation operator, implicit Beam-Warming scheme		airfoils (2-D) and wing (3-D) applications	6	
ECUMB2D	2-D, finite-volume, cell-vertex	multiblock-structured	explicit Lax-Wendroff scheme	local time stepping, multigrid		4, 5	

COMPANY / INSTITUTION: OXFORD UNIVERSITY / Oxford - United Kingdom							
CODE	DISCRET. TECHNIQUE	GRIDS	SPACE/TIME DISCRET.	CONVERGENCE ACCELERATION	SPECIAL FEATURES	REF.	POC
CV_FLOW2	2-D, finite-volume, cell-vertex	single block, O/C-type	central differencing, Jameson-type second and fourth difference dissipation operator, explicit Lax-Wendroff scheme	local time stepping, multigrid	Euler/Navier-Stokes solver	7, 8	3
THREE FLOW	3-D, finite-volume, cell-vertex	single block, H/C-type	central differencing, Jameson-type second and fourth difference dissipation operator, explicit Lax-Wendroff scheme	local time stepping, multigrid	Euler/Navier-Stokes solver	9	

COMPANY / INSTITUTION: SWANSEA UNIVERSITY / Swansea - United Kingdom							
CODE	DISCRET. TECHNIQUE	GRIDS	SPACE/TIME DISCRET.	CONVERGENCE ACCELERATION	SPECIAL FEATURES	REF.	POC
BRITE3D	3-D, finite-volume, cell-vertex	unstructured, adaptation with mesh refinement and remeshing	central differencing, Jameson-type second and fourth difference dissipation operator, multistage Runge-Kutta scheme	local time stepping	Euler/Navier-Stokes solver	10, 11, 12	4
TG	3-D, finite-element	unstructured, adaptation with mesh refinement and remeshing	Taylor-Galerkin finite-element formulation	local time stepping	unsteady 2-D version available	13, 14	
FLITE3D	3-D, finite-element	unstructured with mesh refinement and remeshing	central differencing, explicit, multistage Runge-Kutta,	local time stepping, multigrid	edge based data structure	13-17	

COMPANY / INSTITUTION: NATIONAL AEROSPACE LABORATORY (NLR) / Amsterdam, - The Netherlands							
CODE	DISCRET. TECHNIQUE	GRIDS	SPACE/TIME DISCRET.	CONVERGENCE ACCELERATION	SPECIAL FEATURES	REF.	POC
ENSOLV	3-D, finite-volume, cell-centered	multiblock-structured block-to-block mesh refinement and adaptation	central differencing, Jameson-type second and fourth difference dissipation operator, explicit multistage Runge-Kutta scheme	local time stepping, implicit residual averaging, enthalpy damping, multigrid - full & semi-coarsening	Euler/Navier-Stokes solver, part of CFD system ENFLOW, propeller and jet propulsion options	21-41	5
D2EUL	2-D, finite-volume, cell-vertex	unstructured	Dick-upwind flux difference discretization, MUSCL extrapolation for 2nd order accuracy, explicit multistage Runge-Kutta scheme	local time stepping		42-44	

COMPANY / INSTITUTION: SAAB - SCANIA A.B. / Linköping - Sweden							
CODE	DISCRET. TECHNIQUE	GRIDS	SPACE/TIME DISCRET.	CONVERGENCE ACCELERATION	SPECIAL FEATURES	REF.	POC
MULTEUL	3-D, finite-volume, cell-centered	multiblock-structured	central differencing, Jameson-type second and fourth difference dissipation operator, multistage Runge-Kutta schme	local time stepping, explicit and implicit residual averaging, multigrid	Euler/Navier-Stokes solver	47	6
GEMINI	3-D, finite-volume, cell-centered	structured, automatic grid generation plane by plane	space/time marching scheme, time integration by explicit Runge-Kutta type scheme, space discretization by Roe flux difference splitting upwind scheme	local time stepping, eigenvalue extrapolation technique	Euler/Parabolized Navier-Stokes solver, general missile and aircraft configurations	48	

COMPANY / INSTITUTION: CFD norway as / Trondheim - Norway							
CODE	DISCRET. TECHNIQUE	GRIDS	SPACE/TIME DISCRET.	CONVERGENCE ACCELERATION	SPECIAL FEATURES	REF.	POC
TwoFlow	2-D, finite-volume, cell-centered	multiblock-structured	central differencing, Jameson-type second and fourth difference dissipation operator, explicit 3-stage Runge-Kutta scheme	local time stepping	Euler/Navier-Stokes solver	49	7

COMPANY / INSTITUTION: CFD norway as / Trondheim - Norway							
CODE	DISCRET. TECHNIQUE	GRIDS	SPACE/TIME DISCRET.	CONVERGENCE ACCELERATION	SPECIAL FEATURES	REF.	POC
ThreeFlow	3-D, finite-volume, cell-centered	multiblock-structured	central differencing, Jameson-type second and fourth difference dissipation operator, explicit 3-stage Runge-Kutta scheme	local time stepping	Euler/Navier-Stokes solver, rotating frame of reference option	49	7

3.4.3 Points of Contact

Point of Contact (POC) No. 1:

Code(s): EJ51, EJ50, EJ53, EJ63, EJ65, EJ83
Name: A. Pagano
Dept.: Aerodynamics
Tel.: (+44) 272-693831
Mailing Address: British Aerospace Ltd. (BAe)
Airbus Division
P.O. Box 77
Bristol BS99 7AR
United Kingdom
Tel.-Company: (+44) 272-693831
Fax-Company: (+44) 272-362828
References: 3

Point of Contact (POC) No. 2:

Code(s): ROE/LITTON, HALL/SALMOND, EMJET SERIES, UNSTEADY EULER, ECUMB2D
Name: B. Williams
Dept.: Aerodynamics
Tel.: (+44) 252-392576
Mailing Address: Defence Research Agency (DRA)
Aerodynamics Department
Farnborough
Hants, GU14 6TD
United Kingdom
Tel.-Company: (+44) 252-24461
Fax-Company: (+44) 252-375890
References: 4 - 6

Point of Contact (POC) No. 3:

Code(s): CV_FLOW2, THREE FLOW
Name: K.W. Morton
Dept.:
Tel.: (+44) 865-273885
Mailing Address: Oxford University Computing Laboratory
11 Keble Road
Oxford, OX1 3QD
United Kingdom
Tel.-Company: -
Fax-Company: (+44) 865-273839
References: 7 - 9

Point of Contact (POC) No. 4:

Code(s): BRITE3D, TG, FLITE3D, HYBRID
Name: O. Hassan
Dept.: Aerodynamics
Tel.: (+44) 792-295625
Mailing Address: Computational Dynamics Research
Innovation Centre
University College of Swansea
Singleton Park
Swansea, SA2 8PP
United Kingdom
Tel.-Company: (+44) 792-295625
Fax-Company: (+44) 792-295613
References: 10 - 18

Point of Contact (POC) No. 5:

Code(s): ENSOLV, D2EUL
Name: J.W. Boerstoel, B. Oskam
Dept.: Theoretical Aerodynamics Department
Tel.: (+31) 20-511 3357
Mailing Address: National Aerospace Laboratory NLR
P.O. Box 90502
1006 BM Amsterdam
The Netherlands
Tel.-Company: (+31) 20-511 3113
Fax-Company: (+31) 20-511 3210
References: 21 - 44

Point of Contact (POC) No. 6:

Code(s): MULTEUL, GEMINI
Name: B. Arlinger
Dept.: Aerodynamics
Tel.: (+46) 1318-2583
Mailing Address: SAAB SCANIA A.B.
Saab Aircraft Division
Military Aircraft Sector
581 88 Linköping
Sweden
Tel.-Company: (+46) 1318-0000
Fax-Company: (+46) 1318-1802
References: 47, 48

Point of Contact (POC) No. 7:

Code(s): TwoFlow, ThreeFlow
Name: N. Kubberud
Dept.:
Tel.: (+47) 73-54-0340
Mailing Address: CFD norway as
Teknostallen
Professor Brochsgt. 6
N-7030 Trondheim
Norway
Tel.-Company: (+47) 73-54-0340
Fax-Company: (+47) 73-94-3861
References: 49, 50

3.4.4 References

1. JAMESON A., SCHMIDT, W. and TURKEL, E. "Numerical Solutions for the Euler Equations by Finite Volume Methods Using Runge-Kutta Time Stepping Schemes," AIAA Paper 81-1259, June 1981.

2. JAMESON A. and BAKER T. J. "Multigrid Solution of the Euler Equations for Aircraft Configurations," AIAA Paper 84-0093, 1984.

3. DOE R. H., PAGANO A. and BROWN T. W. "The Development of Practical Euler Methods for Aerodynamic Design," Proc. 15th Conference of the International Council of the Aeronautical Sciences, London, Sept. 7-12, 1986, ICAS-86-1.42.

4. NI R. H. "A Multiple-Grid Scheme for Solving the Euler Equations," AIAA 81-1025, June 1981.

5. HALL M. J. "Cell-Vertex Multigrid Schemes for Solution of the Euler Equations" in: Numerical Methods in Fluid Dynamics II (eds. Morton, K. W. and Baines, M. J.), Oxford University Press, 1986.

6. BEAM R. and WARMING R. F. "An Implicit Factored Scheme for the Compressible Navier-Stokes Equations," *AIAA Journal*, Vol. 16, (1978), pp. 393-402.

7. MORTON K. W. and PAISLEY M. F. "A Finite Volume Scheme with Shock Fitting for the Steady Euler Equations," *Journal of Computational Physics*, Vol. 80, No. 1, (1989), pp. 168-302.

8. MORTON K. W. and RUDGYARD M. A. "Shock Recovery and the cell vertex Scheme for the Steady Euler Equations." In Proc. ICNMFD 11, Springer-Verlag, New York 1989.

9. CRUMPTON P. I. MACKENZIE J. A., MORTON K. W., RUDGYARD M. A. and SHAW G. J. "Cell Vertex Multigrid Methods for the Compressible Navier-Stokes Equations". In Proc. ICNMFD 12, Springer-Verlag, New York 1990.

10. WEATHERILL N. P. and HASSAN O. "Efficient Three-Dimensional Grid Generation Using the Delaunay Triangulation," Proceedings of the 1st CFD Conference, Brussels, Sept. 1992. Pub. Elsevier, Ed. Ch. Hirsch.

11. WEATHERILL N. P., HASSAN O. and MARCUM D. L. "Calculation of Steady Compressible Flow Fields with the Finite Element Method," AIAA Paper 93-0341, Jan. 1993.

12. WEATHERILL N. P., HASSAN O., MARCHANT M. J. and MARCUM D. L. "Adaptive Inviscid Solution for Aerospace Geometries on Efficiently Generated Unstructured Tetrahedral Meshes," AIAA Paper 93-3390, July 1993.

13. PERAIRE J., PEIRO J., FORMAGGIA L., MORGAN K. and ZIENKIEWICZ O.C. "Finite Element Euler Computations in Three Dimensions," *International Journal of Numerical Methods in Engineering*, Vol. 26, (1988), pp. 2135-2159.

14. MORGAN K., PERAIRE J., PEIRO J. and HASSAN O. "The Computation of Three-Dimensional Flows Using Unstructured Grids," *Journal of Computational Methods in Applied Mechanics and Engineering*, Vol. 87, (1991), pp. 335-352.

15. PERAIRE J., MORGAN K. and PEIRO J. "Unstructured Finite Element Mesh Generation and Adaptive Procedures for CFD," Paper presented at the AGARD FDP Specialist"s meeting 24-25, May 1989, Leon, Norway. In "Applications of Mesh Generation to Complex 3-D Configurations," AGARD CP-464, May 1989.

16. PERAIRE J., PEIRO J. and MORGAN K. "A 3D Finite Element Multigrid Solver for the Euler Equations," AIAA Paper 92-0449, Jan. 1992.

17. PEIRO J., PERAIRE J., MORGAN K., HASSAN O. and BIRCH N. "The Numerical Simulation of Flow About Installed Aero-Engine Nacelles Using a Finite Element Solver on Unstructured Meshes," *Aeronautical Journal*, pp. 224-232, 1992.

18. MARCHANT, M. J. and WEATHERILL, N. P. "Adaptivity Techniques for Compressible Inviscid Flows," to appear in : *Jounal of Computer Methods in Applied Mechanics*, 1994.

19. JAMESON, A., BAKER, T. J. and WEATHERILL N. P. "Calculation of Inviscid Transonic Flow Over a Complete Aircraft," AIAA Paper 86-0103, Jan. 1986.

20. JAMESON A. and BAKER T. J. "Improvements to the Aircraft Euler Method," AIAA Paper 87-0452, Jan. 1987.

21. BOERSTOEL J. W. "Preliminary Design and Analysis of Procedures for the Numerical Generation of 3D Block-Structured Grids". Technical Report TR 86102 U, NLR, 1996.

22. BOERSTOEL J. W. "Problem and Solution Formulations for the Generation of 3D Block-Structured Grids, NLR MP 86020 U, 1986.

23. BOERSTOEL J. W. "Preliminary User Guide and Software Design for Aerodynamic Numerical Geometry Specification for Blocked Flow Domains," NLR TR 86103 U, 1986.

24. BOERSTOEL J. W. "Numerical Grid Generation in 3-D Euler Flow Simulation". NLR MP 88013 U (1988), also in: Numerical Methods for Fluid Dynamics, (eds. Morton K.W. and Baines M. J.), Oxford University Press, 1988.

25. JACOBS J. M. J. W., KASSICS A., BOERSTOEL, J. W., BUYSEN F. A. and KUYVENHOVEN J. L. "Numerical Ineractive Grid Generation for 3D-Flow Calculations," NLR MP 88095 U, 1988.

26. BOERSTOEL, J. W., JACOBS J. M. J. W., KASSIES A., AMENDOLA A., TOGNACCINI R. and VITAGLIANO P. L.". Design and Testing of Multiblock Grid Generation Procedure for Aircraft Design and Research". Paper presented at the AGARD FDP Specialist's meeting 24-25 May 1989, Loen, Norway. In "Applications of Mesh Generation to Complex 3-D Configurations," AGARD CP-464, Paper 9, May 1989.

27. BOERSTOEL J. W., SPEKREYSE S. P and VITAGLIANO P. L. "The Design of a System of Codes for Industrial Calculations of Flows Around Aircraft and other Complex Aerodynamic Configurations," NLR TP 92190 L (1992), also: AIAA Paper 92-2619-CP, June 1992

28. SPEKREYSE S. P., BOERSTOEL J. W. and VITAGLIANO P. L. "New Concepts for Multiblock Grid Generation for Flow Domains Around Complex Aerodynamic Configurations," NLR TP 91046 L , 1991, also in : Proc. of the 3rd Int. Conf. on Numerical Grid Generation in Computational Fluid Dynamics and Related Fields, June 1991, (eds. A.S. Archilla, J. Häuser. P.R. Eiseman, J.F. Thompson), North Holland, Amsterdam., 1991, pp. 719-730

29. BOERSTOEL J. W. and SPEKREYSE S. P. "An Information System for the Numerical Simulation of 3D Euler Flows Around Aircraft," NLR TP 90264 L, 1991.

30. AMENDOLA A., TOGNACCINI R., BOERSTOEL J. W. and KASSIES A. "Validation of a Multiblock Euler Flow Solver with Propeller Slipstreams," AGARD CP-437, 1988, Paper P1-1.

31. KUIJVENHOVEN J. L. and BOERSTOEL J. W. "3D Euler Flows Around Modern Airplanes," NLR TR 89343 U, 1989.

32. BOERSTOEL J. W. "Combining Characteristic Forms of Boundary Conditions and Conservation Equations at Boundaries of Cell-Centered Euler-Flow Calculations," NLR MP 87024 U, 1987.

33. JACOBS J. M. J. W., HOEIJMAKERS H. W. M., VAN DEN BERG J. I. and BOERSTOEL J. W. "Numerical Simulation of the Flow About a Wing with Leading-Edge Vortex Flow," NLR MP 88041 U, 1988.

34. JACOBS J. M. J. W. and HOEIJMAKERS H. W. M. "Aspects of the Application of an Euler-equation Method to the simulation of Leading-Edge Vortex Flow," NLR TP 89272 L, 1989.

35. VAN DEN BERG J. I. and BOERSTOEL J. W. "Development and Validation of Characteristic Boundary Conditions for Cell-Centered Euler Flow Calculations," NLR TP 89360 L, 1989.

36. HOEIJMAKERS H. W. M., JACOBS J. M. J. W. and VAN DEN BERG J. I. "Numerical Simulation of Vortical Flow Over a Delta Wing at Subsonic and Transonic Speeds," NLR TP 90029 L, 1990.

37. KASSIES A., TOGNACCINI R. "Boundary Conditions for Euler Equations at Internal Block Faces of Multiblock Domains using Local Grid Refinement," NLR TP 90134 L, 1990.

38. VAN DEN BERG J. I. and BOERSTOEL J. W. "Development and Validation of Characteristic Boundary Condition of a Cell-Centered Euler Method," NLR TP 90144 L, 1990.

39. HOEIJMAKERS H. W. M. and VAN DEN BERG J. I. "Application of an Euler-Equation Method to a Sharp-Edged Delta-Wing Configuration with Vortex Flow," NLR TP 91306 L, 1991, also: AIAA Paper 91-3310, 1991.

40. BUYSEN F. A. and KASSIES A. "Computer Graphics for Fluid Flow Analysis in CAE," NLR TP 89029 L, 1989.

41. BOERSTOEL J. W. "Conceptual Design of a Computer Code System for the Calculation of Flows Around Transport Aircraft," NLR TP 92065 U, 1992.

42. DICK E. "A Flux Difference Splitting Method for the Steady Euler Equations," *Journal of Computational Physics*, Vol. 76 (1988), pp. 19-32.

43. VAN LEER B. "Towards the Ultimate Conservative Difference Scheme," *Journal of Computational Physics*, Vol. 32 (1979), pp. 101-136.

44. DE COCK K. "High-Lift System Analysis Method Using Unstructured Meshes," NLR TP 92351, 1992.

45. HOUTMAN E. M. and BANNINK W. J. "Experimental and Numerical Investigation of the Vortex Flow Over a Delta Wing at Transonic Speed," AGARD CP-494, Paper 5, 1992.

46. KOREN B. "Defect Correction and Multigrid for Efficient and Accurate Computation of Airfoil Flows," *Journal of Computational Physics*, Vol. 77, (1988), pp. 183-206.

47. LÖTSTED P. and SELLEN M. "Multigrid Multiblock Solver of the Stationary Euler and Navier-Stokes Equations," SAAB Report L/O/1 R148, Nov. 1993.

48. SOWA J. "GEMINI - General Missile Navier-Stokes Integrator," ICAS92 Paper 4.3.1, Beijing 1992.

49. ERIKSSON, L. E. and ØRBEKK, E. "Algebraic Block-Strucutured Grid Generation based on a Macro-Block Concept," AGARD CP-464, Paper 3, March 1990.

50. ERIKSSON, L. E. "Simulation of Inviscid Flow Around Airfoils and Cascades based on the Euler Equations," FFA TN 1985-20, 1985.

This page has been deliberately left blank.

Chapter 4

Applications

4.1 AIRFOILS

The Euler equations have been extensively applied to and correlated for the flow over two-dimensional airfoils. The early method developments centered on oscillation-free shock-capturing schemes, either through incorporation of adaptive dissipation coefficients or upwind discretizations. The recent trend has been towards the incorporation of general-geometry adaptive-grid schemes for the treatment of complex configurations, such as the multielement airfoil solutions shown below. These adaptive-grid methods have gained popularity due to their potential to provide highly accurate solutions with relatively few grid points. This gain in popularity owes in large part to the difficulty in modeling complex three-dimensional geometries with globally refined "structured" grids. Several typical solutions over the NACA 0012 airfoil are shown below, as well as several reference (datum) solutions which serve as well-defined test cases for the evaluation of schemes. The computational challenge posed by Pulliam in numerical solutions to the Euler equations for the subcritical flow over a smooth body is also discussed. Finally, an example of the use of Euler equations to scale the results obtained in a gas with nonideal behavior to that of air is reviewed.

4.1.1 NACA 0012 Airfoil

The flow around a NACA 0012 airfoil at M_∞ = 0.8 and α = 1.25 deg is shown in Fig. 4.1.1 using an extension of the 1985 implicit scheme of MacCormack developed at Deutsche Airbus (DA).[1] The pressures indicate an upper surface shock and a weaker lower surface shock. Results include second- and third-order-accurate flux vector splitting formulations together with a second-order treatment of the surface boundary condition. The results demonstrate the sensitivity to discretization and boundary conditions; the first-order results reveal strongly disturbed isoMachlines near the surface of the model over several layers of finite volumes associated with excessive total pressure losses generated in the region of strong acceleration near the leading edge. The pressure distribution is most influenced by the truncation error of the scheme, whereas the entropy error is most influenced by the boundary condition treatment. Third-order accurate spatial discretization and second-order surface boundary condition lead to the entropy variation to be expected, with small losses except for the jump at the upper shock. A small rise of total pressure loss in the field occurs at the transition from subsonic to supersonic flow and is commonly observed in flux vector splitting results.

The flow field and convergence history for the inviscid supersonic flow past a NACA 0012 airfoil (M_∞ = 3, α = 7 deg) is shown in Fig. 4.1.2. All computations were carried out using the same explicit multistage second-order upwind TVD scheme.[2] Standard V-type multigrid cycles were applied. The upwind implicit residual smoothing method (UIRS) is compared to the widely used central implicit residual smoothing (CIRS) method. The CPU times were measured on a single processor CRAY Y-MP. The CIRS method performed best with a (3,2)-scheme (3 stages, 2 dissipation evaluations). The UIRS method performed best with a (3,3)-scheme. The UIRS scheme took more iterations, but the total time to converge 5 orders of magnitude was approximately 35 seconds for either of the schemes with residual smoothing. Further extensions of the residual smoothing concepts are presented by Zhu et al.[3]

4.1.2 Datum Solutions

In 1985 the AGARD Working Group 07 completed the work on the evaluation of numerical results obtained for specified test cases by inviscid flow field methods. The aim of the effort was to compile the state-of-the-art capabilities for computing two- and three-dimensional numerical solutions of the Euler equations for airfoils and wings. Solutions from many researchers of well-known institutions and companies were submitted and subsequently compared with respect to various criteria concerning local and global flow field properties. An evaluation strategy was developed to determine the "best" numerical solution(s) for each test case. Judgments of quality were on the basis of comparisons with known solutions and numerical sensitivities, including grid density, far-field boundary location, and entropy error variations. Datum solutions which can be used as high-quality reference solutions for the comparison of methods are presented in Ref. 4. Two transonic cases from that study are shown in Figs. 4.1.3–4.1.6. Recent adaptive-grid computations for a third test case from that study are shown in Fig. 4.1.7; these more recent computations resolve the large spread in the originally contributed results.

NLR 7301: M_∞ *= 0.721,* α *= –0.194 deg*

As a remarkable example of the solution accuracy which could be obtained at that time, the Mach number contours from the numerical results of Schmidt/Jameson[4] for their two-dimensional computation around the NLR 7301 airfoil section at "shock-free" design conditions are presented in Fig. 4.1.3. Euler computations around the NLR 7301 profile with the specified flow conditions pose a special difficulty as an "exact" hodograph solution by Boerstoel and Huizing[5] exists which indicates the development of a supercritical shockless flow. Many contributors were unable to compute a shockless flow and, therefore, failed to predict accurate global forces and moments. The shock-free solution is an isolated design point, and slight variations in geometry or angle of attack will cause a single or double shock to appear on the airfoil. The

shock-free solution was generated from a hodograph solution which, unfortunately, is not without some discretization error, as particularly evidenced in the pressure variation near the occurrence of the first sonic point. The airfoil itself is only defined discretely at a large number of points. It is difficult to retain the shock-free feature as the grid is refined.[6] There is some question whether this is due to: slight errors in the hodograph solution, or the extrapolated infinite-grid solution being computed on a geometry which is inconsistent with a shock-free case because of limitations in the geometry specification. The accuracy of the solution is probably best judged by monitoring for convergence of the global forces and moments as the grid is refined uniformly with well-posed boundary conditions and by checking for inconsistencies in the solution, such as entropy losses upstream of shocks.

Two numerical solutions, namely those by Rizzi and Schmidt/Jameson, were considered as numerical reference solutions according to their high quality.[4] In Fig. 4.1.5, the surface distributions of pressure and total pressure loss of Schmidt/Jameson are depicted together with the curves according to the "exact" hodograph solution (E). The overall agreement is very good except at the leading edge, where a disagreement is noticed in the representation of the suction peak. This defect seems to be systematic in nature and can be found in all numerical solutions presented to the working group. Also, minor deviations from the hodograph solution are evident on the upper and lower surfaces wherever stronger pressure gradient changes occur. Considering the total pressure loses, all values are well below 1 percent. With the exception of the suction peak region and the area around the midchord lower side, very small or almost zero values underline the high numerical standard achieved.

Source	Mesh	C_L	C_D	C_M
Rizzi	160x32 O-type	0.597	0.0002	-0.130
Jameson /Schmidt	322x66 O-type	0.594	0.0005	-0.132
Exact		0.594	0.0005	-0.130

Table 4.1.1 Aerodynamic coefficients for NLR 7301 test case.

The forces and moments from the two datum solutions are compared to the exact solution in Table 4.1.1; the variations are very small. A full discussion of the variations between eighteen solutions for global forces and moments are summarized in Ref. 4, including useful information regarding solution features, such as mesh extent and grid topology and density.

NACA 0012: $M_\infty = 0.85$, $\alpha = 1$ *deg*

Fig. 4.1.4 shows the Mach number distribution from the datum solution of Schmidt/Jameson.[4] An analysis of all seven contributions concerning global forces and moments is provided in Ref. 4; there was a relatively high scatter of 16 percent in lift and 36 percent in moment coefficient. The reasons for that behavior are attributed to the uncertainty in the determination of the shock positions. The outer boundary distance and numerical properties (such as artificial viscosity, truncation error, or convergence) were important influencing factors.

The results of Salas/South and Schmidt/Jameson were selected as the best two contributing solutions.[4] The coefficients from these two datum solutions are shown in Table 4.1.2 and indicate close agreement. Both solutions are plotted in Fig. 4.1.6 as surface pressure and total pressure loss distributions. The latter solution shows extremely small total pressure variations over almost all the upper and lower side, with the exception of some minor excursions at the shock positions. The errors in both methods are small enough that either can serve as numerical reference solutions.

Source	Mesh	C_L	C_D	C_M
Salas /South	192x39 O-type	0.3472	0.0557	-.1167
Jameson /Schmidt	320x64 O-type	0.3584	0.0580	-.1228

Table 4.1.2 Aerodynamic coefficients for NACA 0012 test case; $M_\infty = 0.85$ and $\alpha = 1$ deg.

NACA 0012: $M_\infty = 0.95$, $\alpha = 0$ *deg*

This test case, originally considered by the AGARD working group,[4] is characterized by an oblique shock structure emanating from the trailing edge, with a weak normal shock in the wake as shown in Fig. 4.1.7. The contributed computations showed wide variations in the predicted location of the sonic point on the downstream chord-line extension, from 1.4 to 3.1 chords downstream from the trailing edge, associated with the large cell sizes downstream of the airfoil.

The normal shock is relatively weak with a Mach number of less than 1.1 ahead of the shock. An analysis indicates that the location of the downstream normal shock wave is very sensitive to the resolution of the expansion waves in the supersonic zone above the airfoil, since that sets the oblique shock angle at the trailing edge.[7] The normal shock location is quite sensitive to small errors because the length of the oblique shock emanating from the trailing edge to the shock triple point is about five chord lengths.

The correct location of the normal shock downstream of the trailing edge has been determined through a grid convergence study[7] performed using a structured-grid code. The grids utilized include 65 x 25, 129 x 49, 257 x 97, and 2049 x 765 O-type grids. The effect of grid density on the location of the normal shock is shown in Fig. 4.1.7, where the shock location is measured downstream of the trailing edge. The infinite-grid normal-shock location obtained in this manner is about 3.35 chords from the trailing edge. Corresponding mesh-refinement results using the GAUSS2D method of Hartwich[8] are also shown and indicate a similar shock location of approximately 3.32.

The ability of adaptive methods to obtain accurate results is examined using two different Euler solvers in Ref. 7.

The two solvers corresponded to upwind-biased methods: a fully unstructured-grid method using triangular cells and a semiunstructured-grid method using quadrilateral cells. The adapted grids from these two methods are shown in Fig. 4.1.7. The adaptive grids demonstrate that the shock features can be resolved well at large distances from the airfoil since small cell sizes can be maintained all along the shock waves. The accuracy, however, is determined more by the resolution of the smooth portions of the flow than by the resolution of the more prevalent flow features, such as discontinuities. The results presented used an adaption criteria which led to consistent results in normal-shock position as the number of points is increased. The adaption criteria used repaired inconsistencies in several commonly used methods of adaption.

4.1.3 Multielement Airfoils

The capability to compute flows over complex geometries is extremely important to the aerodynamic designer. As alternatives to block-structured grid methods, the unstructured grid methods have greatly expanded the capabilities in that direction using both Euler and Navier-Stokes equation sets. These methods are natural as a framework for the accommodation of arbitrary geometries and the incorporation of adaptive meshing techniques.

Results from the computations of Mavriplis and Jameson[9,10] for a two-element airfoil (slat and main airfoil) are shown in Fig. 4.1.8. The method uses triangular and adaptive grids, and the results shown were some of the first which demonstrated the power of the unstructured grid method in aerodynamic applications. A node-based central-difference finite-volume scheme is used, which has been shown to be equivalent to a Galerkin finite-element method, with a lumped-mass-matrix term.[9,11] Several of the computational grids which are used in the solution process are shown in the top half of Fig. 4.1.8; the grids shown are the finest adapted mesh and three of the coarser grids used in the mesh sequencing and multigrid acceleration processes. The grids have been generated independently of each other, thus decoupling the grid generation process from the multigrid acceleration scheme. Finer grids can be obtained by global refinements or by adapting previous coarser grids; the grids communicate through an efficient tree-search algorithm. The lower half of the figure shows the pressures on the slat/airfoil and the convergence rate of the algorithm using single and multiple grids. The pressures have been compared elsewhere[9] to experimental and potential results for unadapted meshes at slightly different conditions with generally good results. The convergence rate shown is comparable to that attained in structured-grid multigrid codes and substantially improved over that attainable with only a single grid; comparisons (not shown) of the unadapted and adapted grid solutions indicate that higher accuracy can be obtained with fewer points in the latter approach.

The flow over a three-element NLR 422 airfoil computed with an unstructured finite-volume, vertex-centered code[12] is shown in Fig. 4.1.9. A second-order-accurate upwind flux-difference-splitting algorithm[13] is solved to steady state using an explicit Runge-Kutta scheme. The far-field and near-field views of the grid indicate a gradual enlargement of the grid away from the surface. The grids are generated in an automatic way, driven by overall user-defined parameters, like maximum allowable grid spacing and curvature. The pressure distributions over the slat, main element, and flap at a subsonic condition (M_∞ = 0.20, α = 10 deg) indicate excellent correlation with a potential flow method. This particular case was chosen in order to make a comparison of Euler computations with a potential flow result. Other adaptive-grid computations for a similar case are given by Dimier and Ronzheimer.[14]

4.1.4 6:1 Ellipse

Numerical solutions of the Euler equations obtained for subsonic airfoils with sharp trailing edges have faithfully reproduced the expected solutions obtained for viscous flow, namely smooth flow at the trailing edge. Euler solutions have, with grid refinement, returned those of potential theory with a Kutta condition imposed at the trailing edge. For a smooth trailing edge, Pulliam[15] has presented a currently unresolved problem: the inviscid subcritical flow over a 6:1 ellipse section. The potential flow solution is unique up to the specification of the circulation or, equivalently, the rear stagnation point on the surface. Assuming irrotational initial and boundary conditions, the inviscid solution should remain irrotational. However, the discrete Euler solutions obtained by Pulliam with a central-difference finite difference solution and by several others with different formulations return an arbitrarily large value of lift for any combination of grid and/or angle of attack which is nonsymmetric. The resulting solutions, however, agree closely with potential flow solutions with a circulation imposed equal to that obtained in the discrete Euler solutions, as shown in Fig. 4.1.10. The Euler solutions are sensitive to solution parameters such as grid refinement and stretching, boundary conditions, dissipation coefficients, convergence level, etc. However, no consistent explanation of the discrepancy obtained with the different discrete Euler solution could be found. Winterstein and Hafez[16] show the numerical interplay of the dissipation forms and the boundary conditions for blunt bodies. For viscous algorithms which use Euler formulations as a building block for the convective and pressure terms, the issue is not considered to be relevant as the viscous separation at the trailing edge serves to set the lift coefficient.

4.1.5 Heavy-Gas Airfoil Computations

The capability to conduct three-dimensional wind-tunnel tests at full-scale Reynolds number has long been an important aerodynamic need. Full-scale tunnels are only available at low speeds because of size and power constraints; high Reynolds number ground-based testing is generally achieved through combinations of high pressure, cryogenic temperatures, or alternate test gases. The cryogenic temperature approach enables Mach number to be varied independently of Reynolds number, but testing is expensive and complex. Heavy gases, such as sulfur hexafluoride (SF_6), are an attractive alternative to air because of the increased Reynolds number available due to lower viscosity and increased density, for fixed length and velocity. In addition, power consumption and model loads are less than that for air at a constant Mach number. Unfortunately, most heavy gases behave as nonideal gases, and the results obtained have to be correlated with those for air, since the ultimate objective is the performance of the tested configuration in a high-Reynolds number air flow.

The difficulty introduced by the nonideal gas behavior is shown in Fig. 4.1.11; the pressure distribution over a NACA 0012 airfoil at M_∞ = 0.8 and α = 1.25 deg is shown for air and for

a heavy gas, SF_6, at various pressures.[17] The air results would be the same independent of the pressure since air behaves as an ideal gas. The pressure distributions shown are quite different because of the nonideal behavior of the SF_6. As the pressure is increased, the upper surface shock moves forward relative to air and the lower surface shock disappears. A numerical scaling procedure was developed[17] based on the use of the Euler equations and the transonic small disturbance equations. Using the transonic scaling procedure in which an equivalent gamma is determined based on both temperature and pressure, the pressure distributions are shown in Fig. 4.1.11 for air and for SF_6 at various combinations of pressure and temperature. Note the Mach number for the SF_6 calculations is different from that in air in order to match the transonic similarity parameter. The procedure closely correlates inviscid results in SF_6 to those in air. The viscous scaling between air and SF_6 introduces additional complications, especially at transonic speeds; the inviscid scaling procedure used above leads to different shock locations in viscous flow at high Reynolds numbers. Further details and references to earlier works in this area are given in Ref 17.

4.1.6 References

1. RILL, S., "A High-Resolution, Implicit Scheme for the Solution of the Euler Equations," DGLR-Paper 89–167, 1989, pp. 377–386.

2. BLAZEK, J., KROLL, N., RADESPIEL, R. and ROSSOW, C.-C., "Upwind Implicit Residual Smoothing Method For Multi-Stage Schemes," AIAA 91–1533, 1991.

3. ZHU, Z., LACOR, C. and HIRSCH, H., "A New Residual Smoothing Method for Multigrid, Multistage Schemes," AIAA 93–3356–CP, 1993.

4. AGARD FLUID DYNAMICS PANEL WORKING GROUP 07, "Test Cases for Inviscid Flow Field Methods," AGARD Advisory Report No. 211 (ISBN 92–835–1497–1), 1985.

5. BOERSTOEL, J. W. and HUIZING, G. H., "Transonic Shock Free Aerofoil Design by Analytic Hodograph Methods," NLR MP 73 023 U; also AIAA 74–539, 1974.

6. RAJ, P., OLLING, C. R., and SINGER, S. W., "Applications of Multizone Euler/Navier-Stokes Aerodynamic Methods to Aircraft Configurations," ICAS-90–6.4.4, Sept. 1990.

7. WARREN, G. P., ANDERSON, W. K., THOMAS, J. L. and KRIST, S. L., "Grid Convergence for Adaptive Methods," AIAA 91–1592–CP, June 1991, pp. 729–741.

8. HARTWICH, P. M., "Split Coefficient Matrix (SCM) Method with Floating Shock Fitting for Transonic Airfoils," *Lecture Notes in Physics*, Vol. 371, Springer-Verlag, New York, 1990, pp. 394–399.

9. MAVRIPLIS, D. and JAMESON, A., "Multigrid Solution of the Euler Equations on Unstructured and Adaptive Meshes," ICASE Rept. 87–53, July 1987; also NASA Contractor Report 178346.

10. MAVRIPLIS, D., "Solution of the Two-Dimensional Euler Equations on Unstructured Triangular Meshes," Ph.D. Thesis, Princeton University, 1987.

11. JAMESON, A., BAKER, T. J. and WEATHERILL, N. P., "Calculation of Inviscid Transonic Flow Over a Complete Aircraft," AIAA 86–0103, Jan. 1986.

12. DE COCK, K., "High-Lift System Analysis Method Using Unstructured Meshes," NLR TP 92351, 1992.

13. DICK, E., "Multigrid Formulation of Polynomial Flux Difference Splitting for Steady Euler Equations," Journal of Computational Physics, Vol. 91, Nov. 1990, pp. 161–173.

14. DIMIER, A. and RONZHEIMER, A., "Solution of the Euler Equations on Unstructured and Adaptive Meshes," DLR-IB 129-90/44, 1990.

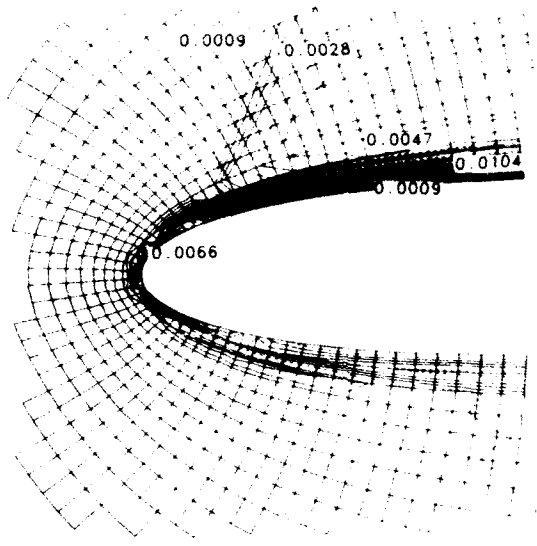
15. PULLIAM, T., "Computational Challenge: Euler Solutions for Ellipses," AIAA Journal, Vol. 28, No. 10, Oct. 1990, pp. 1703-1704.

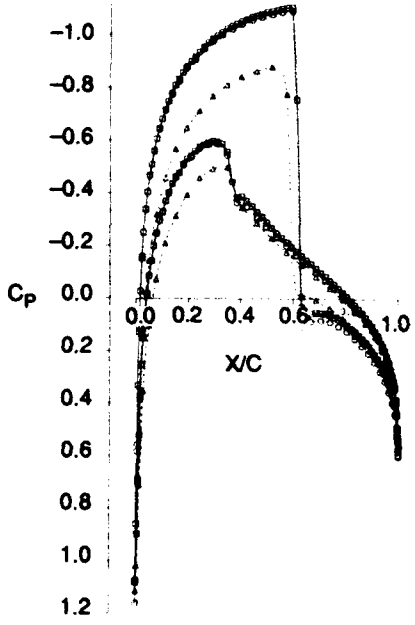
16. WINTERSTEIN, R. and HAFEZ, M., "Euler Solutions for Blunt Bodies Using Triangular Meshes: Artificial Viscosity Forms and Numerical Boundary Conditions," AIAA 93–3333–CP, July 1993.

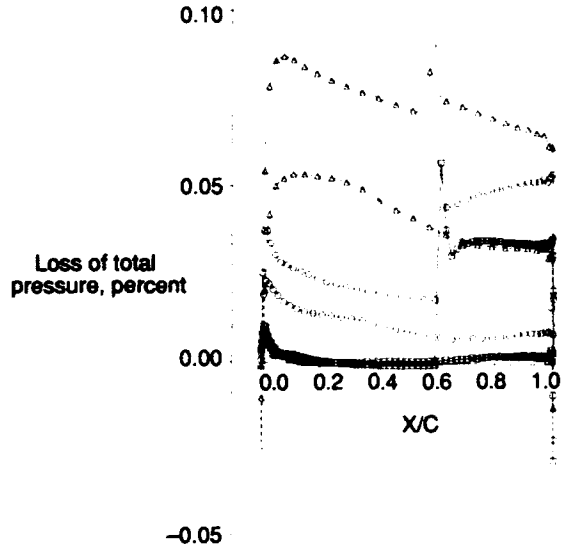
17. ANDERSON, W. K., "Numerical Study of the Aerodynamic Effects of Using Sulfur Hexafluoride as a Test Gas in Wind Tunnels," NASA TP–3086, May 1991.

Third-order 2D-Implicit Euler Code
Second-order boundary condition, 160 × 32 points

Computational grid and lines of constant total pressure loss near the leading edge (higher resolution)

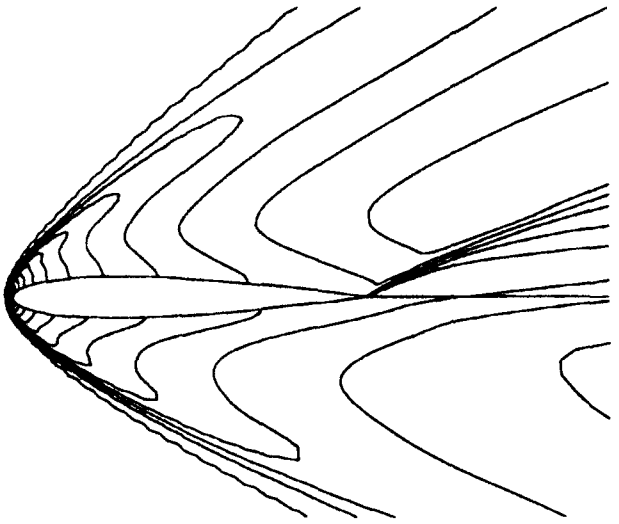






- -△- - 1st-order 2D Implicit Euler Code, 160 × 32 points
  1st-order boundary condition C_L = .240, C_D = 0.031
- ···o··· 3rd-order 2D Implicit Euler Code, 160 × 32 points
  1st order boundary condition C_L = .330, C_D = 0.021
- —□— 3rd-order 2D Implicit Euler Code, 160 × 32 points
  2nd-order boundary condition C_L = .340, C_D = 0.022

Figure 4.1.1 Sensitivity of Euler results to algorithm dissipation and boundary condition for NACA 0012 airfoil; M_∞ = 0.8, α = 1.25 deg.



M_∞ = 3, angle of attack = 7°

	CFL	Residual smoothing	CPU, sec
1	0.9	none	92
2	16-24	upwind implicit	34
3	2.5	central implicit	38

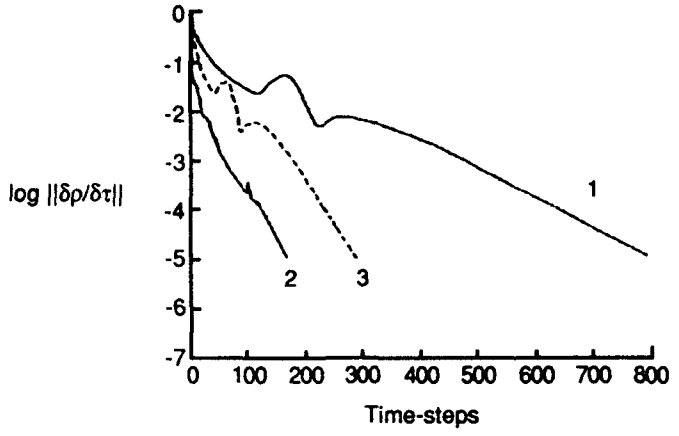
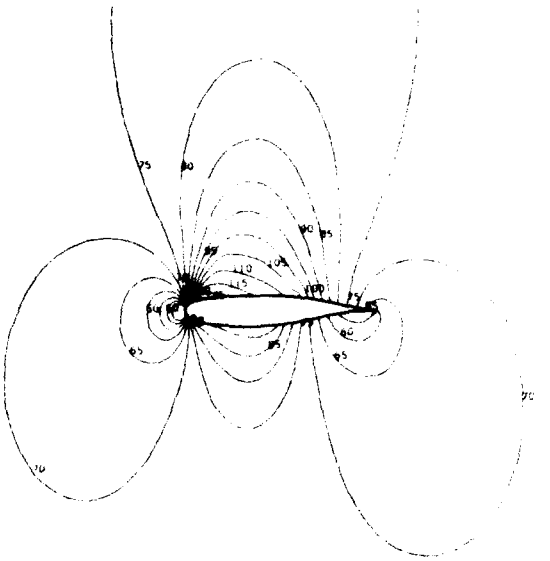
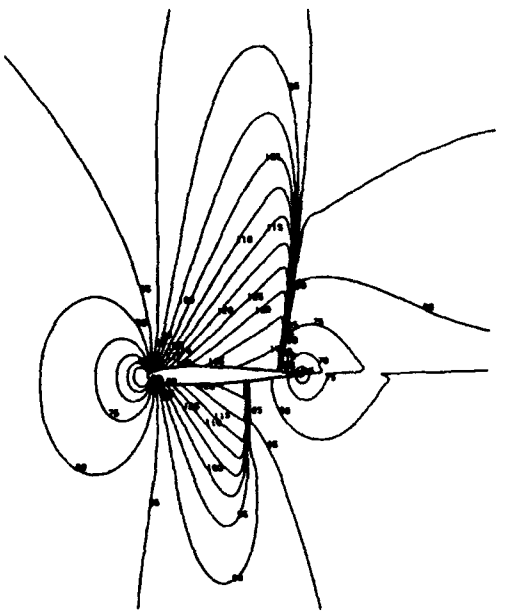


Figure 4.1.2 Mach contours and convergence history for the supersonic flow past a NACA 0012 airfoil; M_∞ = 3, α = 7 deg.



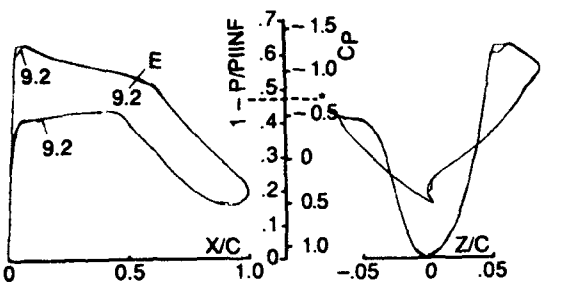
Iso-Mach lines of solution n° 9.2
(ΔM = 0.05, curve parameter = 100M).

Figure 4.1.3 Mach contours for NLR 7301 datum solution; M_∞ = 0.721, α = –0.194 deg.

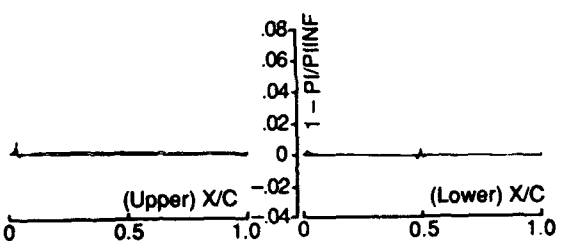


Iso-Mach lines of solution n° 9
(ΔM = 0.05, curve parameter = 100M).

Figure 4.1.4 Mach contours for NACA 0012 datum solution; M_∞ = 0.85, α = 1.0 deg.



Pressure distribution



Total pressure distribution
M = 0.721, α = 0.194

Figure 4.1.5 Comparisons of datum and hodograph solutions for NLR 7301 airfoil; M_∞ = 0.721, α = –0.194 deg.

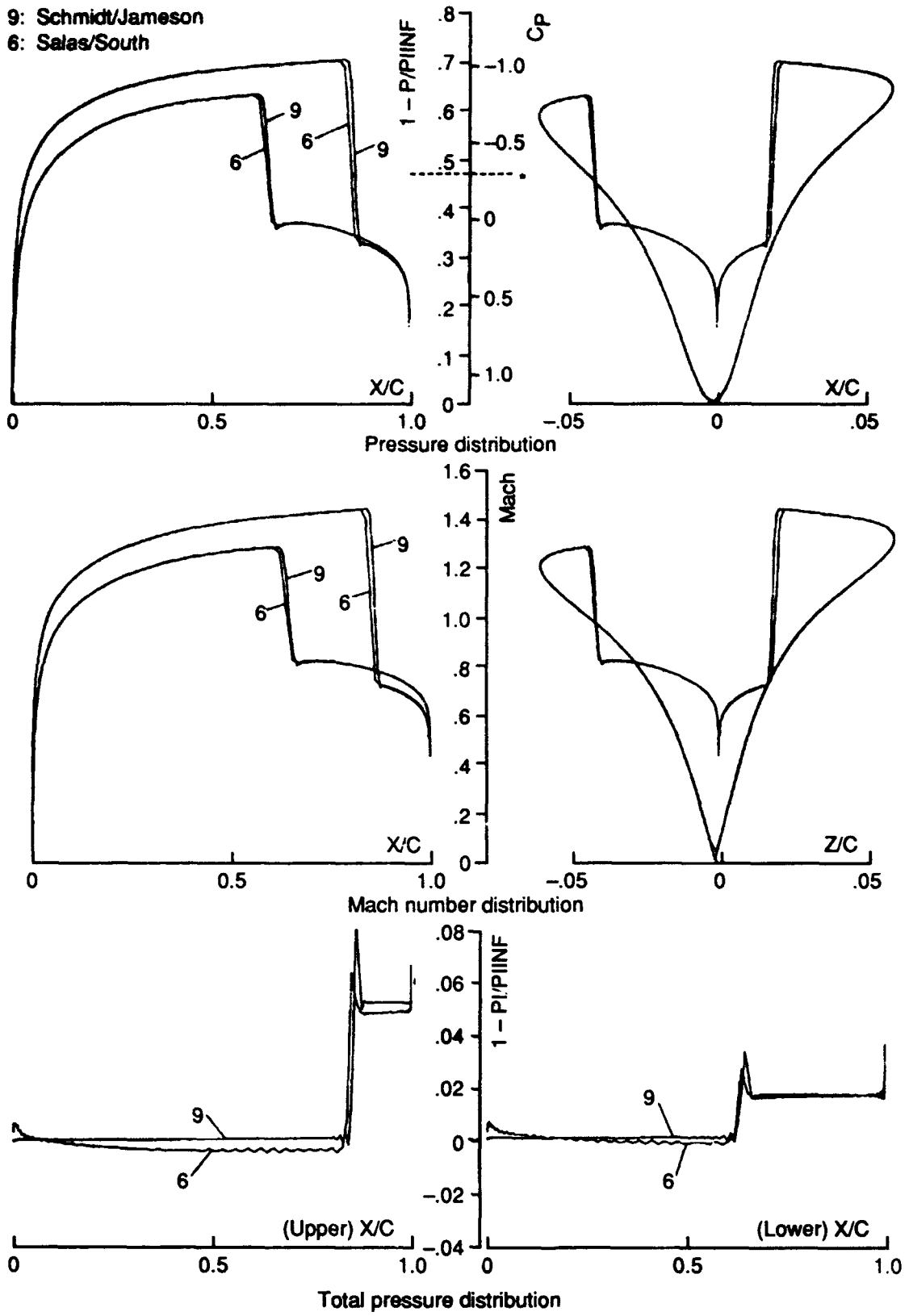
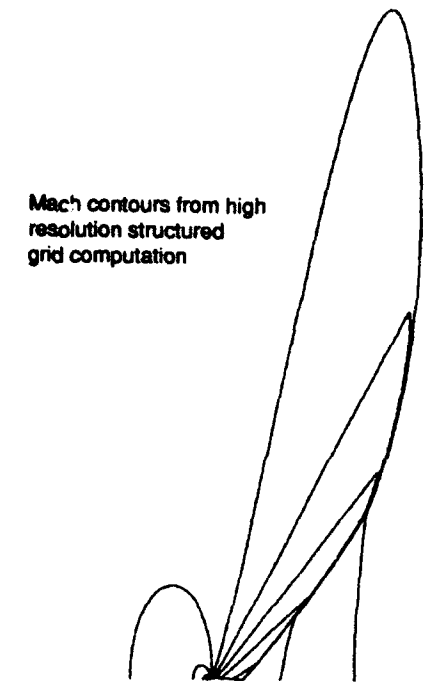
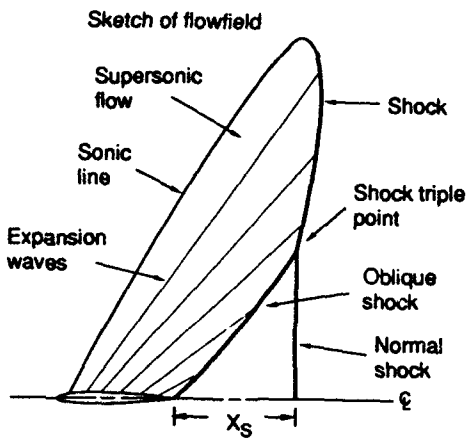
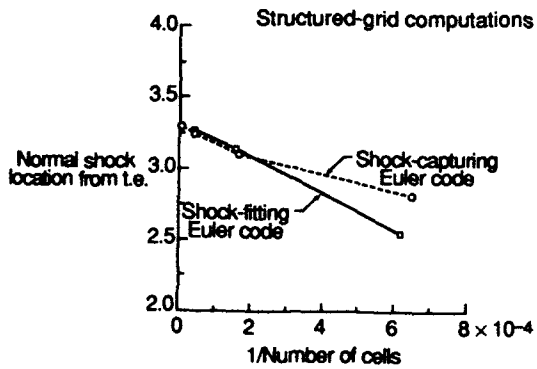
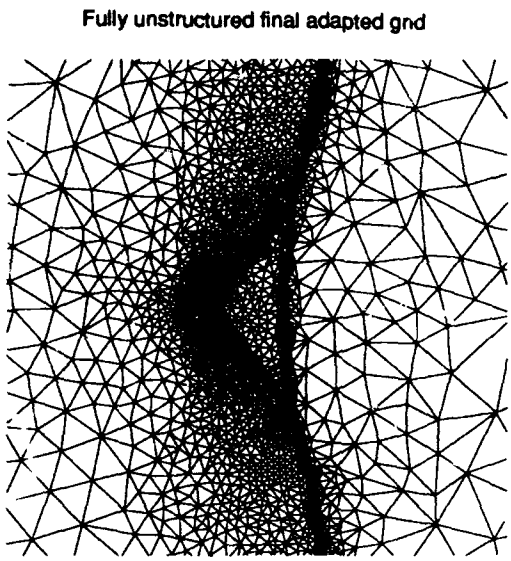


Figure 4.1.6 Comparisons of two datum solutions for NACA 0012 airfoil; $M_\infty = 0.85$, $\alpha = 1.0$ deg.









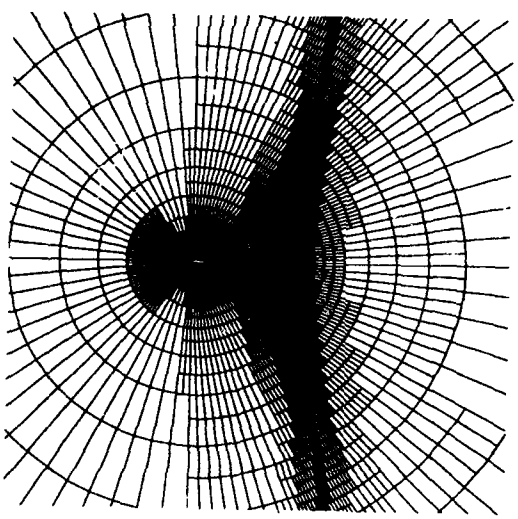


Figure 4.1.7 Adaptive-grid Euler computations for NACA 0012 airfoil; $M_\infty = 0.95$, $\alpha = 1.0$ deg.

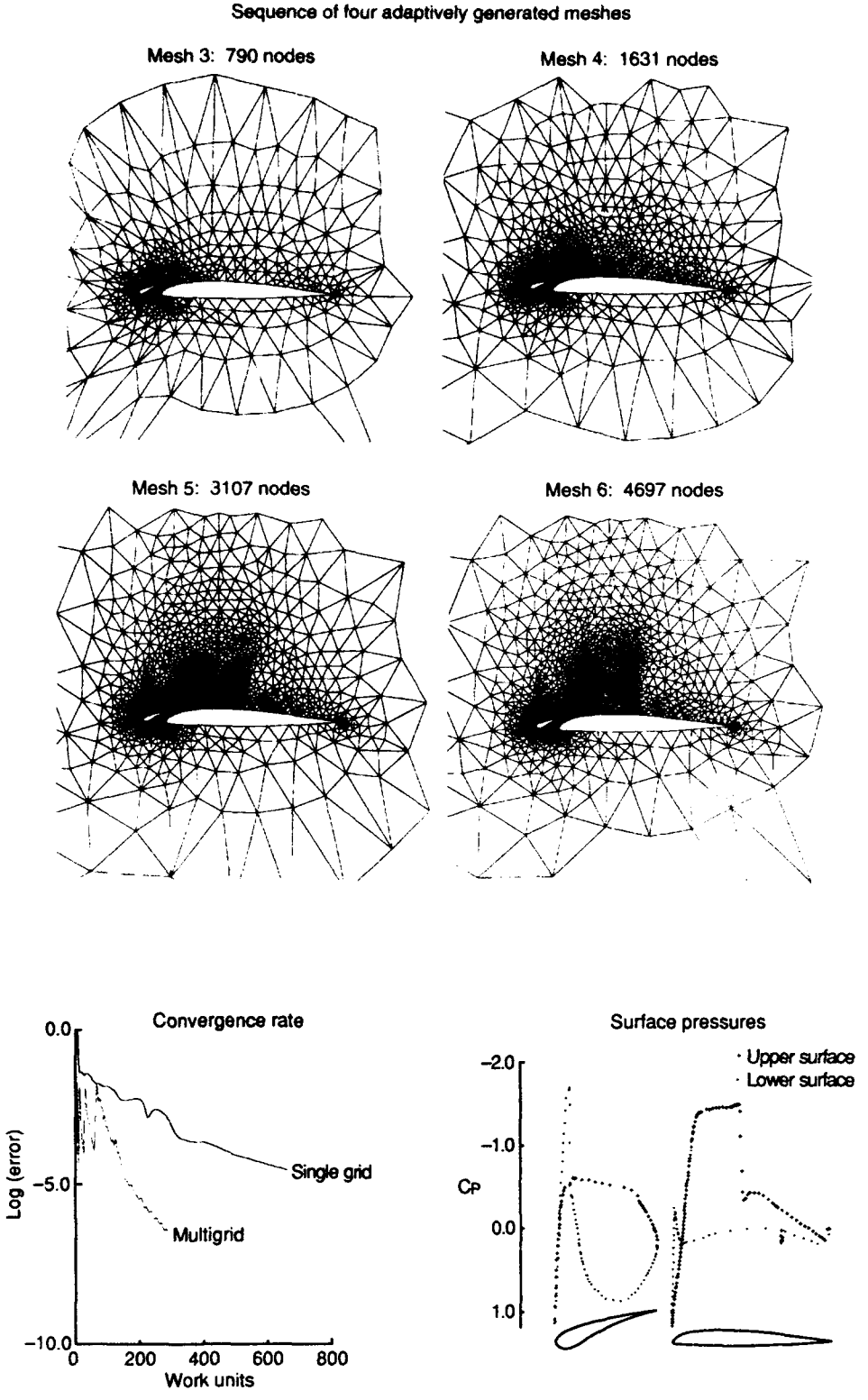
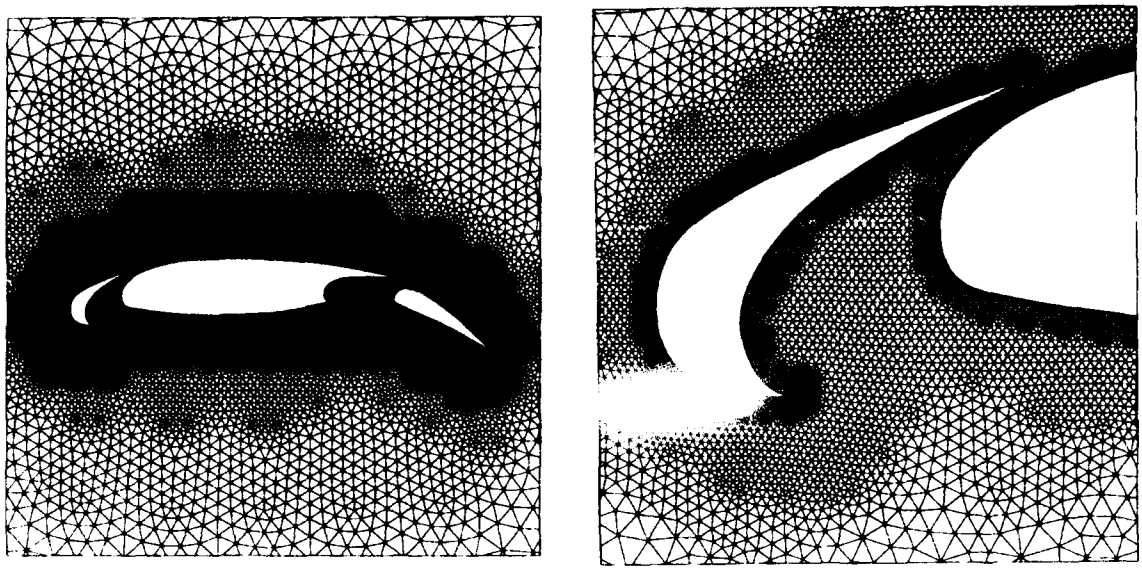


Figure 4.1.8 Adaptively–refined grids, computed pressures, and convergence rate for unstructured-grid computations of a two–element airfoil (main airfoil and slat); M_∞ = 0.7, α = 2.8 deg.



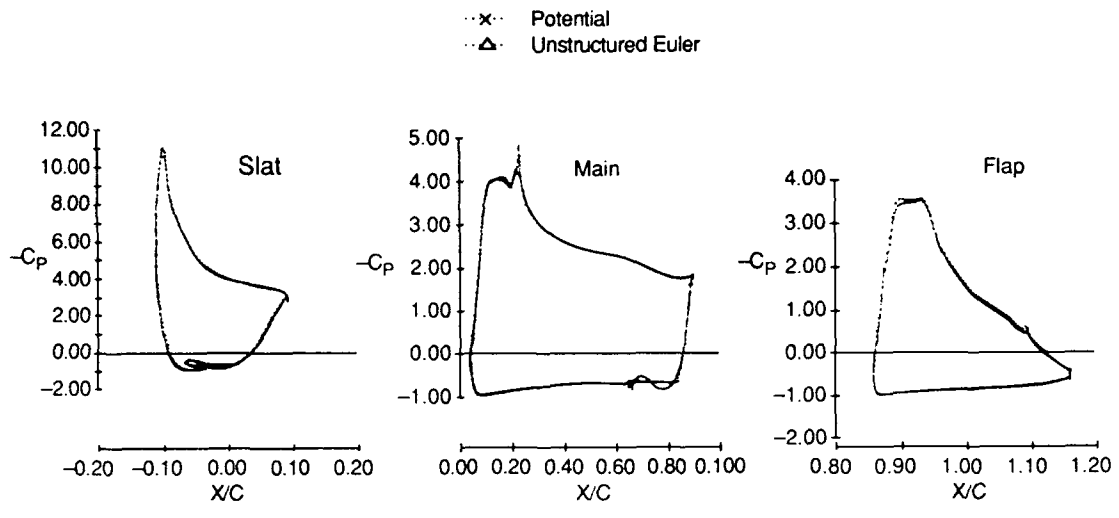


Figure 4.1.9 Unstructured-grid Euler method applied to multielement airfoil; $M_\infty = 0.2$, $\alpha = 10.0$ deg.

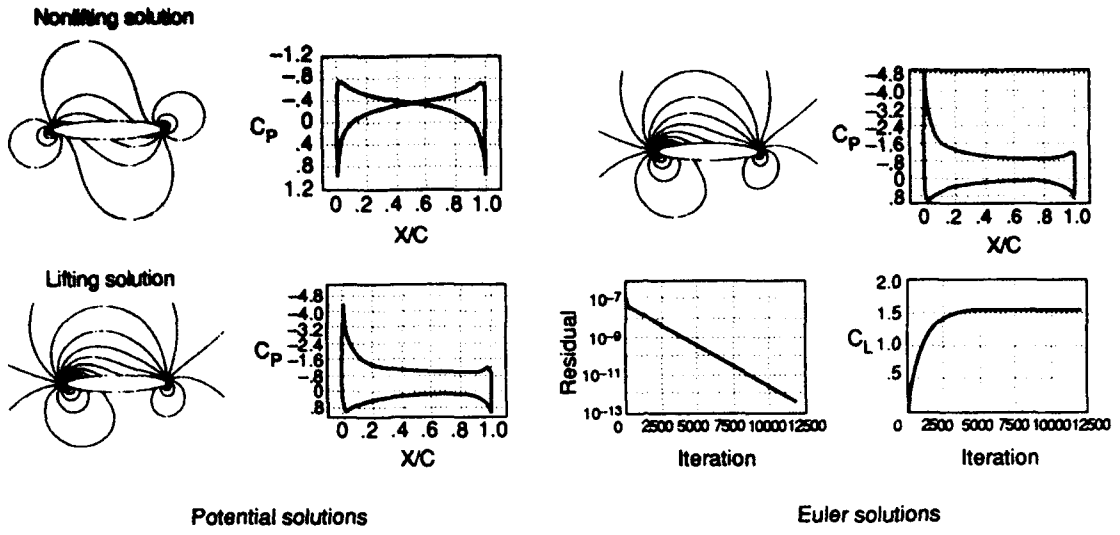


Figure 4.1.10 Inviscid subcritical flow about a 6:1 ellipse from analytic solutions to the potential equations and numerical solutions to the Euler equations; α = 0.0 deg.

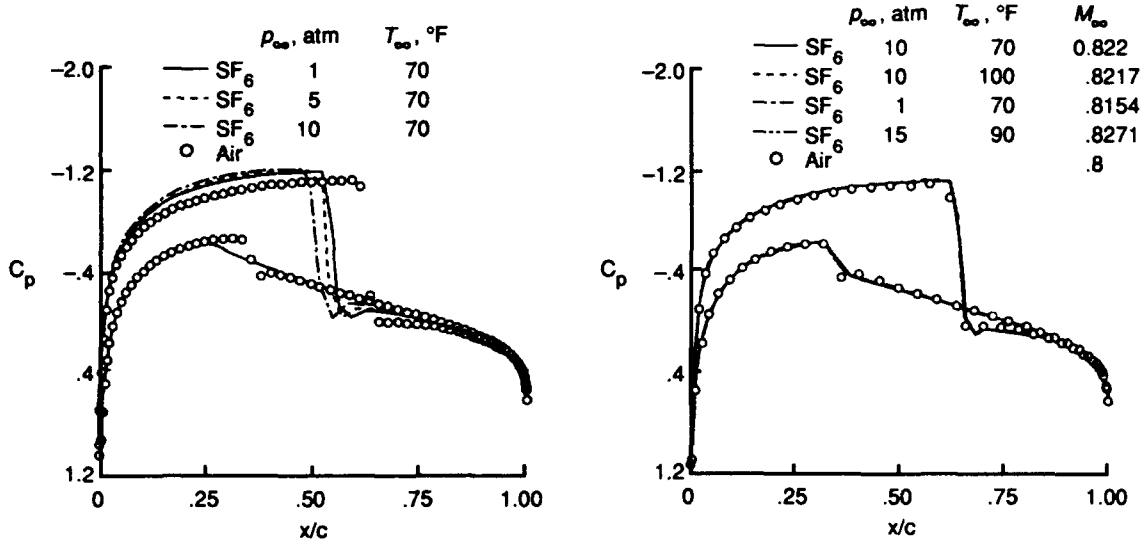


Figure 4.1.11 Effect of non-ideal gas behavior on the pressure distribution for a NACA 0012 airfoil; M_∞ = 0.8, α = 1.25 deg.

4.2 WINGS

Applications are shown below of Euler codes applied to attached flows over wing components. These types of computations can be done routinely with modern computers and algorithms. Without the modeling of viscous terms, the solutions to the Euler equations generally overestimate the experimental lift and, at transonic speeds, produce shock locations which are generally too far aft of high Reynolds number experimental data. Calculations for the ONERA M6 wing are shown because it is a widely used test case for the comparison of Euler codes, using both structured and unstructured grids. The other three examples are comparisons of the capabilities of different codes for a transport-type wing and two lower aspect-ratio fighter-type wings. The examples include comparisons between different Euler methods and with irrotational methods, such as nonconservative and conservative potential methods.

4.2.1 ONERA M6 Wing

The ONERA M6 wing at transonic conditions (M_∞ = 0.84 and α = 3.06 deg) has been used extensively as a test case for the verification of Euler methods. The results shown in Fig. 4.2.1 are typical of those that are attainable with finite-volume codes, in this case an upwind-biased implicit scheme[1] using the flux-vector splitting of Van Leer. At these conditions, the predicted shock is slightly aft of the measured experimental data, obtained at a Reynolds number of 11.7 million.[2] The pressures are shown for a 97x17x17 C-O mesh and a 193x33x33 C-O mesh; the differences between the two structured meshes are limited to the regions of the upper surface shocks. The surface pressures are shown at the right of the figure for a C-O mesh and a C-H mesh; the resolution of the C-O mesh at the tip is considerably better and leads to a greatly improved resolution of the pressures in the tip region.

In Fig. 4.2.2, the grid and Mach contours from the unstructured cell-vertex method of Mavriplis[3] is shown. The grid is adapted to the shock patterns on the wing and plane of symmetry. For the nonadaptive case, the lift coefficient on meshes of 9,428 cells, 53,961 cells, and 357,900 cells was 0.2713, 0.2872, and 0.2923. With adaption, the lift coefficient was 0.2901 with 173,412 cells, close to that of the finest mesh with better resolution in the shock and leading-edge regions. The full multigrid method is used to accelerate convergence of the scheme, with an order of magnitude benefit in efficiency.

The results of Mavriplis demonstrated that complex geometries could be solved on unstructured grids with similar multigrid performance to that attained with structured grids. The computer resources for the scheme were somewhat higher (approximately 100 words/vertex and 75–100 microseconds/multigrid-cycle/vertex on the CRAY-YMP computer) compared with the structured-grid codes (40 words/cell and 30 microseconds/multigrid-cycle/cell on the CRAY-YMP computer). The additional resources required by the unstructured-grid scheme are offset by the ease in which the grid can be adapted to the local features of interest.

4.2.2 DFVLR F4 Wing

In order to increase the understanding of the prediction accuracy for the flow around general aerodynamic configurations, a detailed study was made under the auspices of the European GARTEUR Action Group AG05 regarding the prediction capabilities of a number of different codes for the flow over a simple wing.[4] The wing selected was the DFVLR F4 wing, shown in Fig. 4.2.3 at two conditions: one subsonic (M_∞ = 0.30, α = 0.84 deg) and one transonic (M_∞ = 0.75, α = 0.84 deg). Ten methods were applied, which included nonconservative and conservative potential methods and three Euler methods. Studies were made and reported of convergence characteristics, grid refinement, and grid extent. A few selected results from the study are shown.

For the subsonic case, the global lift and drag predictions increase from nonconservative potential to conservative potential to Euler; somewhat surprisingly, the variations also decrease in the same manner, although some of this effect may be because the three Euler solvers were closely related central-difference schemes. The largest discrepancies occurred outboard on the wing, principally due to different modelings of the wing tip. The pressure distributions from all of the contributions at a semispan location of 0.821 are shown and indicate "essentially the same pressure distribution in a fairly narrow band," thus lending high confidence to their validity in predicting pressure variations due to configurations changes.

The global lift and drag coefficients for the transonic case, however, indicate substantially larger differences than for the subsonic case. The nonconservative potential codes predicted the shock position too far forward in relation to conservative potential or Euler codes, attributed to the lack of mass conservation in the former class of schemes. The predictions from the latter two classes are shown at the semispan location of 0.821 and indicate that the variations in lift coefficients are largely attributed to variations in shock position. Additional computations for this case includes the embedded grid computations of Radespiel[5] and the grid refinement computations of Leicher.[6] As pointed out in the reference,[4] the results indicate only a "snapshot" of capability existing in 1986; advances in computational capability have occurred since that time which have allowed greater confidence for engineering computations of transonic flows with the Euler equations.

4.2.3 RAE Wing-Fuselage

Accurate prediction of transonic and supersonic wave drag is critical in fighter aircraft design optimization. The flows are complex and highly nonlinear; the Euler equations, although neglecting viscosity, with consequent errors in shock position and strength, can be used to predict overall configuration effects including rotational flows due to shock curvature or free vortices. Three different Euler solvers were compared for transonic flow field computations on a wing-fuselage configuration by Agrawal et al.[7] The three schemes compared were the explicit central-difference FLO67 code, the implicit upwind-biased code CFL3D, and a nonconservative upwind code ET2. The three codes were compared on two configurations and were evaluated with respect to accuracy and convergence. A sample calculation is shown in Fig. 4.2.4 for the transonic flow over an RAE wing-fuselage geometry at M_∞ = 0.9 and α = 1 deg. The results predicted by the three methods are similar, except for differences in coarse-grid regions and near shocks. The upwind finite-volume code predicted shock waves with the best resolution and was least sensitive to grid refinement. The best convergence was obtained with the central-difference

FLO67 code using local time stepping and full multigrid acceleration. The latter code was particularly robust in this range of application; its predictions generally suffice for flows near design conditions where shock/boundary-layer interaction effects are small.

4.2.4 F5 Wing

As part of a code validation study,[8] a number of transonic computations were made using the implicit upwind Euler code CFL3D and the transonic small-disturbance code CAP-TSD for both steady and unsteady applications. The small-disturbance code incorporated both entropy and vorticity corrections, thereby extending its applicability into regimes where shock strength normally precludes its use. The purpose of the evaluation was to determine the accuracy and applicability of the methods by performing detailed studies to assess the influence of several parameters in the numerical modeling of the solution. The F-5 wing was used as a test case; it has a panel aspect ratio of 1.58, a leading-edge sweep angle of 31.9 deg, and a taper ratio of 0.28. The calculations are compared with the experimental pressure data from Tijdeman et al.[9] Unsteady comparisons are shown in a subsequent section.

Three-dimensional steady flow computations at M_∞ = 0.95 and α = 0.0 deg are shown in Fig. 4.2.5. Comparisons between the Euler and potential flow pressures shown indicate excellent correlation for all three span stations. Along the upper surface of the wing, there is a mild shock wave that is predicted by both the Euler and potential codes, although it is not evident in the experimental data. Euler computations were made using three grids with approximately (1) 1,000,000 ;(2) 250,000; and (3) 140,000 grid points. The finest grid and the medium grid resulted in essentially identical results.

4.2.5 References

1. ANDERSON, W. K., THOMAS, J. L. and WHITFIELD, D. L., "Three-Dimensional Multigrid Algorithms for the Flux-Split Euler Equations," NASA TP-2829, Nov. 1988.

2. SCHMITT, V., and CHARPIN, F., "Pressure Distributions on the ONERA-M6-Wing at Transonic Mach Numbers," AGARD AR-138, May 1979, pp. B1-1-B1-44.

3. MAVRIPLIS, D., "Three-Dimensional Unstructured Multigrid for the Unstructured Grid Euler Equations," AIAA 91-1549-CP, June 1991, pp. 239-247.

4. CARR, M. P., "Accuracy Study of Transonic Flow For Three-Dimensional Wings," AGARD CP 347, Vol. 1, Paper No. 18, 1988.

5. RADESPIEL, R., "Efficient Solution of Three-Dimensional Euler Equations Using Embedded Grids," Proc. of 15th ICAS Congress, Paper No. 86-1.3.3., London, UK, September 1986.

6. LEICHER, S., "Transonic Wing Study of the DLR-F4 Wing," AGARD CP-412, Paper No. 14, 1986.

7. AGRAWAL, S., LOWRIE, R. B. and CREASMAN, S. F., "An Evaluation of Euler Solvers for Transonic Computations on Wing-Fuselage Configurations," AIAA 90-3015, August, 1990.

8. ANDERSON, W. K. and BATINA, J. L., "Accurate Solutions, Parameter Studies, and Comparisons for the Euler and Potential Equations," Paper No. 15, AGARD CP-437, Vol. 1, May 1989.

9. TIJDEMAN, H., VAN NUNEN, J. W. G., KRAAN, A. N., PERSOON, A. S., POESTKOKE, R., ROOS, R., SCHIPPERS, P. and SIEBERT C. M., "Transonic Wind Tunnel Tests on an Oscillating Wing with External Stores," AFFDL-TR-78-194, Dec. 1978.

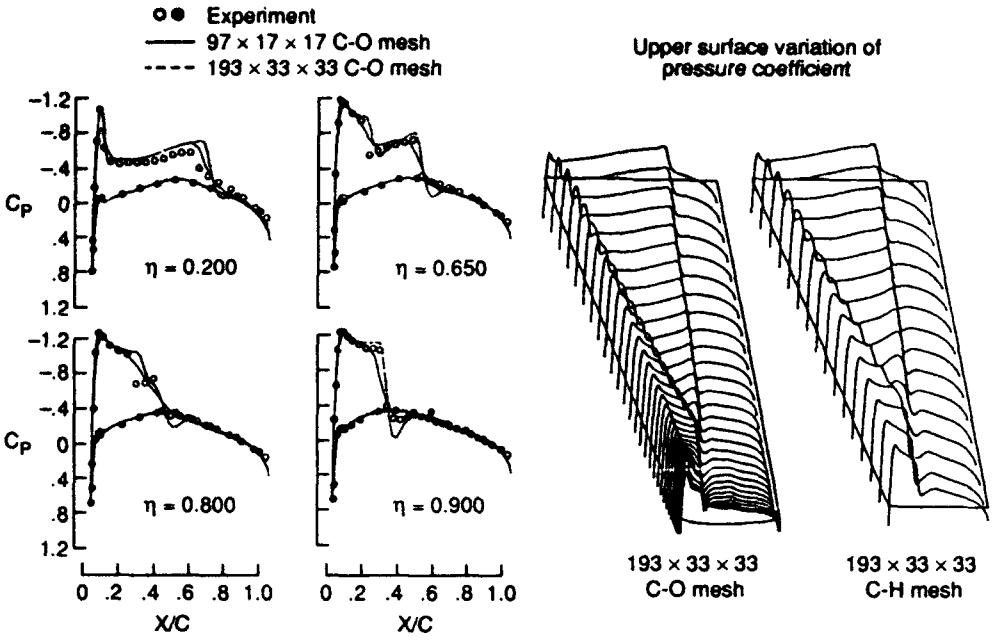


Figure 4.2.1 Comparison of structured-grid Euler computation with experiment for ONERA M6 wing; $M_\infty = 0.84$, $\alpha = 3.06$ deg.

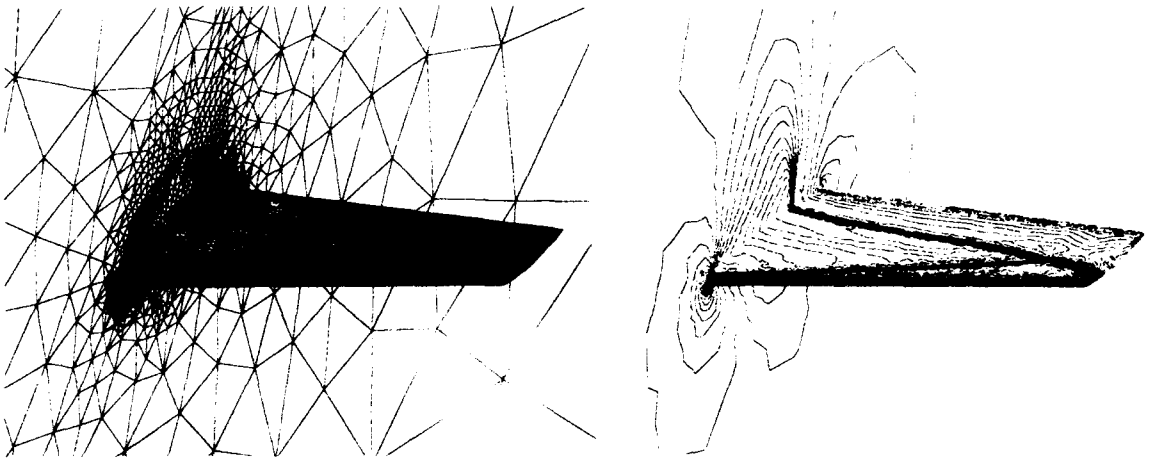
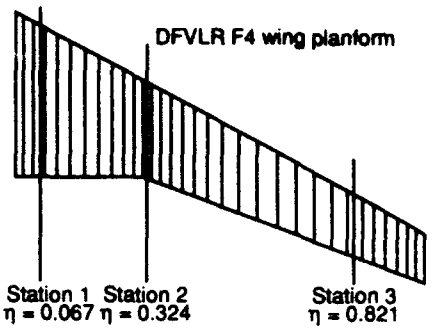
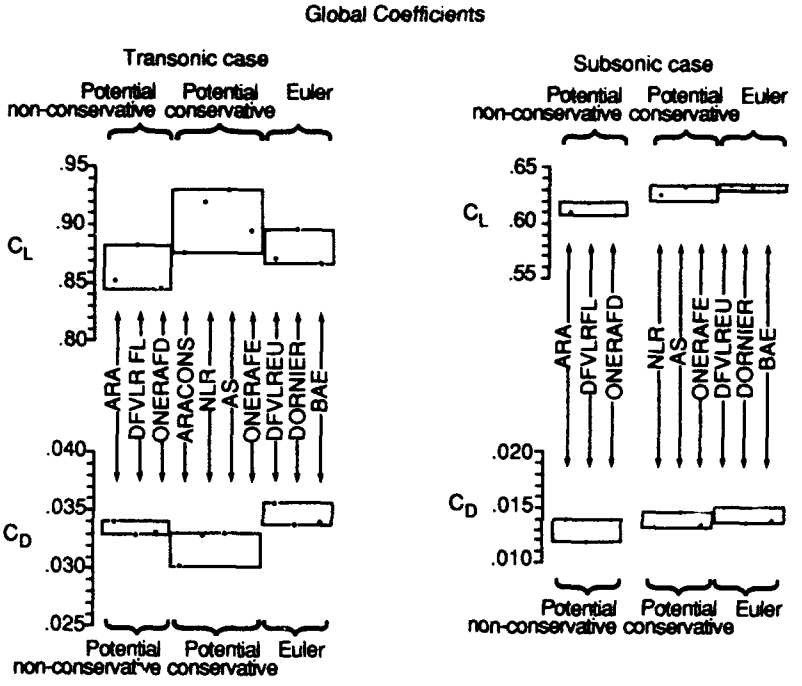


Figure 4.2.2 Adaptive, unstructured grid (left) and Mach contours (right) on surface and in plane of symmetry for ONERA M6 wing; $M_\infty = 0.84$, $\alpha = 3.06$ deg.





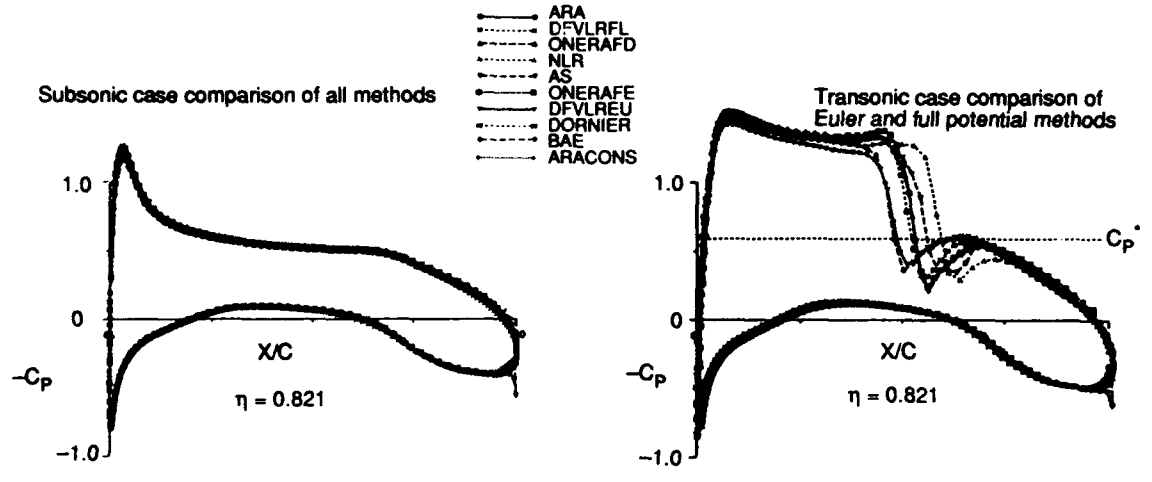


Figure 4.2.3 Comparison of Euler computations for DFVLR F4 wing planform.

(a) Pressure distribution at 0.60 semispan location.

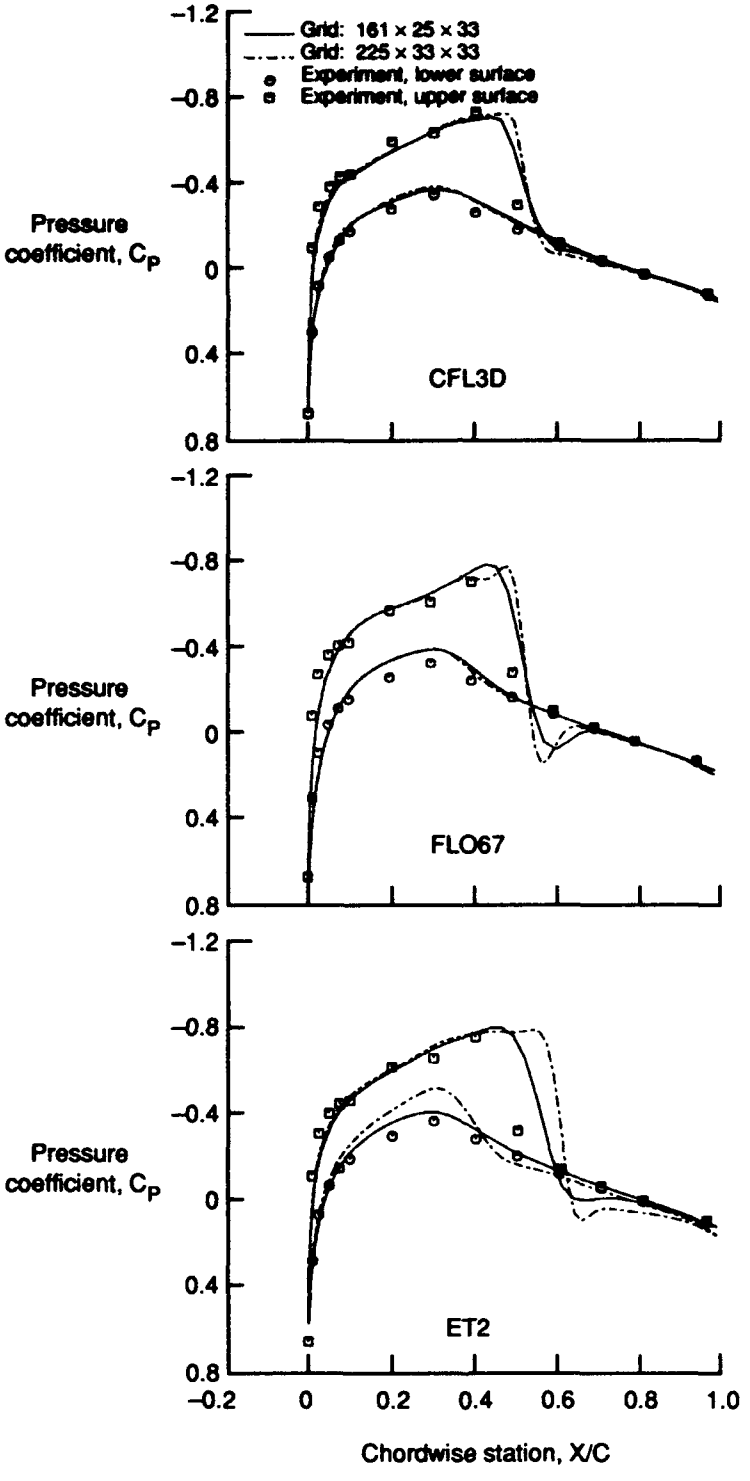
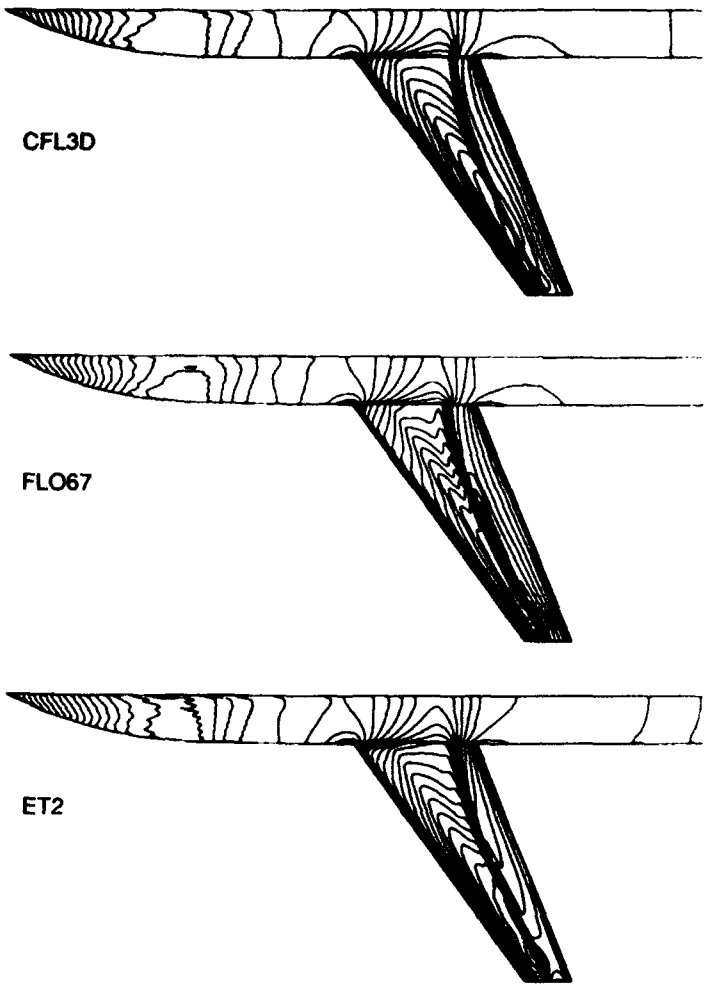


Figure 4.2.4 Sectional and surface pressures on RAE wing-fuselage; $M_\infty = 0.90$, $\alpha = 1$ deg.

(b) Surface pressure contours.



Grid: 225 × 33 × 33

Figure 4.2.4 Concluded.

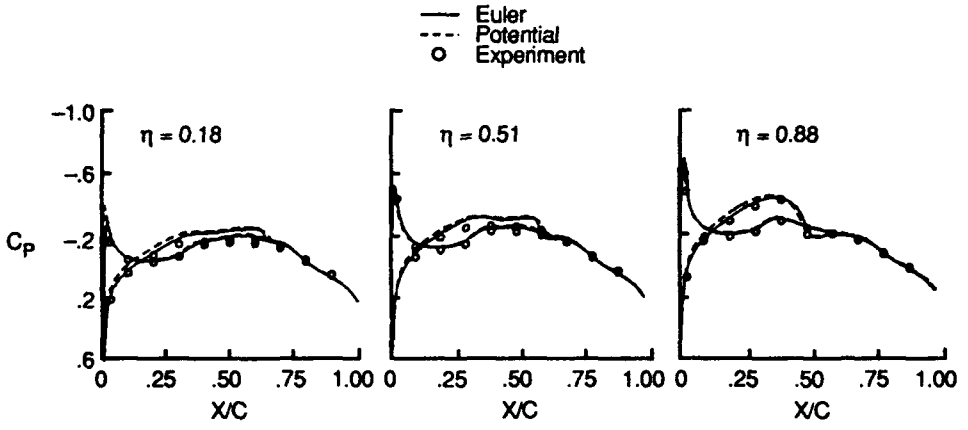


Figure 4.2.5 Comparison of steady pressure distributions for the F-5 wing; M_∞ = 0.9, α = 0.0 deg.

4.3 VORTICAL FLOWS

Vortical flows play an important role in the high-angle-of-attack aerodynamic performance of aerodynamic configurations, especially contemporary military aircraft. The growth, interaction, and decay of these vortices are highly nonlinear with angle of attack and are difficult to predict accurately with numerical methods. Current design practices rely extensively on experimental test programs.

For thin lifting surfaces, the Euler equations can model the principal features of the interactions, in that primary separations arising from sharp leading edges can be predicted; sufficient artificial viscosity is introduced to faithfully model the effects of the true viscosity. Secondary separation arising from boundary-layer separations, as well as from primary separation arising from round leading edges, is common in practice and must be modeled by incorporation of boundary-layer effects in the Euler model or by considering the Reynolds-averaged Navier-Stokes equations. The solutions from the Euler equations can be regarded as limit solutions of the viscous equations at high Reynolds numbers, in which boundary-layer interactions are small. These solutions, accounting for the principal interactions, can be especially useful for configuration effects. For bodies, the primary vortices are shed from flow separations on smooth surfaces; thus, some form of viscous modeling (either a boundary-layer or prescribed-separation model) must be incorporated for the Euler equations to be of use.

4.3.1 Delta Wings

The flows over simple delta and double-delta wings have been studied extensively, both experimentally and computationally. The overall physical structure of the subsonic flowfield over a low aspect ratio delta wing at angle of attack is well understood.[1-4] The characteristics of the flow field are dominated by the two counterrotating primary vortices which form over the wing because of separation along the leading edges. The flow reattaches close to the leeward symmetry plane of the wing; as the flow proceeds outboard, it experiences an adverse pressure gradient, leading to a secondary separation-induced vortex. The secondary vortex can in turn lead to a tertiary vortex underneath and inboard of the secondary vortex. The influence of turbulence is to delay the secondary separation to a more outboard position and generally to eliminate any tertiary separation.

At supersonic speeds, the now classical work of Stanbrook and Squire[5] revealed that the boundary between attached and separated flow patterns could be classified readily in terms of Mach number and angle of attack, both measured normal to the leading edge. Their original experimental work was extended further in a number of experimental studies[6-8] to identify regions associated with shock-induced separations, separation bubbles, and crossflow shocks. McMillin et al.[9] performed a systematic computational investigation of the parametric experiments of Miller and Wood,[8] including computations with both Euler and Navier-Stokes algorithms. The flow classification of Miller and Wood was refined near the boundary between attached and separated flows based on a reexamination of the experimental data in the light of the additional insight obtained with the computational methodology; the resulting classification is shown in Fig. 4.3.1. For Mach numbers normal to the leading edge less than 1, the variation of the separated flow patterns with angle of attack is similar to that found at subsonic speeds. For supersonic normal Mach numbers, the flow transitions with increasing angle of attack from attached flow with an inboard crossflow shock to separated vortical flow with an inboard crossflow shock located above the vortex.

For supersonic flows, the reduction in computational work associated with the conical equations has been used to advantage in computations for vortical flows over conical bodies and wings, both for Euler and Navier-Stokes equations. With an adaptive conical-flow solver, Powell et al.[10,11] has studied extensively the total pressure losses in vortical flows simulated by the Euler equations over sharp leading edges. An example is shown in Fig. 4.3.2 for the flow over a 75-deg swept delta wing at a Mach number of 1.95 and 10 deg angle of attack. The pitot pressures computed with an adaptive, central difference solver for the conical Euler equations are compared to the experimental results of Monnerie and Werle[12] at a Reynolds number of 0.95 million, based on root chord. The flow field induced by the primary vortex is very similar between the Euler calculation and experiment. The secondary separation underneath the primary vortex leads to a secondary vortex which is not modeled in the Euler equations; a crossflow shock, rather than a secondary separation, is predicted under the primary vortex.

McMillin et al.[9] have made extensive comparisons with both Euler, laminar, and turbulent flow models for parametric variations in wing sweep, angle of attack, and Mach number. A comparison of pressure between Euler, laminar Navier-Stokes, and experiment is shown in Fig. 4.3.3 for sweeps of 75, 67.5, and 60 deg over a range of angles of attack. The predominant features of the experiment are captured by the Euler calculations, especially as the angle of attack is increased. Based on these and other parametric computations, McMillin et al.[9] produced the envelope of conditions, shown in Fig. 4.3.4, where Navier-Stokes and Euler solutions give similar results for the primary vortex flow structure. The envelope corresponds to regions where the flow is either clearly separated at the leading edge (through a subsonic leading-edge condition or a high angle of attack) or regions where the flow is clearly attached (through a supersonic leading-edge condition at low angle of attack). At intermediate angles of attack for supersonic Mach numbers, both measured normal to the leading edge, the vortical/separated flow structures lie close to the surface and shows a marked sensitivity to the viscous model. In fact, as the Mach number was increased, the laminar flow computations tended to agree best with the nominally turbulent experiments of Miller and Wood, believed to be associated with an incomplete transition to turbulent flow at the model trailing edge for the higher Mach numbers. In all cases, the Euler computations were incapable of predicting any of the secondary flow features such as secondary vortices or separation induced by a shock.

4.3.2 Delta Wing in Yaw

The adaptive method of Powell et al.[10,11] has been applied to a series of delta wings tested parametrically by Miller and Wood, including effects of flap deflection and yaw. The general trends of the experiments are well predicted. An example is shown in Fig. 4.3.5 for the flow over a 75-deg swept delta wing in sideslip (M_∞=1.7, α=12 deg, β=8 deg). The refinement possible with the adaptive mesh in the region of the leading

edges is evident. The grid corresponds to an equivalent mesh of 256 x 128. The pressures show an asymmetrical flow pattern, corresponding to a stronger shallower vortex on the port side relative to a weaker more circular vortex on the starboard side. The pressures agree well with experiment, although both vortices are predicted too far outboard and the suction levels are overpredicted.

4.3.3 Double-Delta Wing

Longo and Das[13] numerically investigated the vortical flow past a double-delta wing at moderate angles of attack. For this study, the DLR multiblock Euler solver CEVCATS was used. The selected configuration is a thin flat plate double-delta wing with sweep angles of 80 deg and 60 deg for the strake and the wing, respectively. A body-fitted mesh with an O-O topology is used to discretize the physical domain. It contains 56 cells in the chordwise direction, 112 cells in the spanwise direction, and 48 cells in the normal direction. Solutions are evaluated for the subsonic case $M_\infty = 0.30$ at $\alpha = 7$ and 11 degs by correlating with available experimental data.[14]

The numerical results in Fig. 4.3.6 indicate that at $\alpha = 7$ deg (sequence of left side figures), two vortices are formed on each side of the wing which originate from the wing apex and from the wing leading-edge kink. Downstream of the leading-edge kink, the strength of the inner vortex decreases because it is no longer fed with vorticity. On the other hand, the whole vorticity shed from the rear part of the leading edge is fed into the outer vortex. Thus, its strength increases downstream. The two vortices with the same sense of rotation tend to move around each other. Since the outer vortex is stronger, the tendency leads to an outward and downward movement of the inner vortex. At $\alpha = 11$ deg (sequence of right side figures), the two vortices merge over the wing. Due to the influence of the outer vortex, the weaker inner vortex moves outwards and joins to the outer vortex. At the wing trailing edge, only a single vortex can be identified. The Euler solution predicts the general changes of the vortices structure in good agreement with the experimental data. Discrepancies between computed and measured location of the merging vortices are due to the neglect of secondary vortices on the numerical solution.

4.3.4 Vortex Breakdown

For delta wings, the maximum lift generally occurs at angles of attack above that corresponding to the onset of vortex breakdown at the trailing edge of the wing; with increasing sweep, the maximum lift becomes coincident with the occurrence of vortex breakdown at the trailing edge. Vortex breakdown computations have been made by Agrawal et al.[15] using Euler and both laminar and turbulent Navier-Stokes equations for a 70–deg swept delta wing. Streamwise vorticity contours in the pre- and post-breakdown regions of the wing at α=30 deg are shown in Fig. 4.3.7. The experimental data obtained using Laser Doppler Velocimetry (LDV) is compared to Euler and laminar, turbulent, and embedded laminar Navier-Stokes computations. The predicted contour levels in the primary vortex are similar, although peak levels are much higher in the experiment, which can be attributed to the diffusion associated with the numerical truncation of the scheme. Secondary vortex structures are evident in the viscous computations and experiment, as expected. The vorticity levels in the post-burst position are much smaller, in relation to pre-burst levels, for both computation and experiment. The computations are asymmetric with respect to the vortex core, attributed to the inability of any of the computations to model the increased levels of turbulence associated with the burst vortex region, as shown by LDV data. The progression of vortex breakdown position with angle of attack for both Euler and Navier-Stokes computations was shown to be consistent with the experimental data. The turbulent flow calculations showed breakdown upstream of both laminar and Euler computations and, in general, at angles of attack where breakdown approached the apex of the wing, either viscous calculation showed significantly improved agreement with experimentally observed breakdown locations. Hitzel[16–18] presents extensive studies of the vortex breakdowns computed with the Euler equations for a swept delta wing.

4.3.5 IEPG Vortex Flow Computation/Experiment

To assess the capabilities of computational methods for simulating the flow around a typical military aircraft planform, a collaborative program among four nations (United Kingdom, Germany, Italy, and the Netherlands) was started in 1987 under the auspices of the Independent European Programme Group (IEPG).[19] The isolated sharp-edged cropped delta wing planform shown in Fig. 4.3.8 was modeled through computations with the Euler and Navier-Stokes equations and compared to results of an experiment conducted by Elsenaar and Hoeijimakers.[20] The section of the wing is an NACA 64A005 section which is blended into a bi-convex section ahead of the maximum thickness location. The case selected was a transonic flow case (M_∞=0.85, α=10 deg) corresponding to a full-span leading-edge vortex flow with weak shocks only. Vortex breakdown over the wing occurs experimentally at angles of attack greater than 20-22 degs.

The IEPG effort is actually a follow-on to an earlier International Vortex Flow Experiment on Euler Code Validation,[21,22] which was conducted from 1983–1987 with the express intent of obtaining and comparing detailed experimental data, especially at transonic speeds, for a 65–deg swept cropped delta wing with existing computational methods. The participating organizations included the FAA, NLR, AFWAL, DLR, MBB, DORNIER, and the technical universities at Delft and Braunschweig. The experimental and computational data base form this rather successful program is summarized in Ref. 21 and the entire program is reviewed in Ref. 22. The planform chosen for the IEPG effort was similar to that in the earlier effort; the experimental data base was enlarged, especially in the transonic range, and computations included both Navier-Stokes and Euler solvers.

Euler solutions were obtained using a single body-fitted grid of approximately 300,000 points with seven different Euler codes. The dissipation coefficients were reduced as much as possible within the constraint of obtaining a converged solution. The results indicated that the pressure distributions were predicted reasonable closely for all of the methods; the differences were most noticeable in the apex, leading-edge, and trailing-edge areas. There were significant differences between the various total pressure predictions attributed to the magnitude of the truncation errors on a single, fixed grid. However, the correlations of the Euler solutions with experimental data show much larger differences because of the neglect of the boundary-layer-induced secondary separation effects in the

Euler method. The lift coefficient differences between the predictions was 5 percent; the average Euler-predicted lift values were 12 percent larger than in the experiment.

The surface pressure contours and lateral pressures at two longitudinal stations are shown in Fig. 4.3.8. Only two of the seven Euler solutions, which are representative, are compared to results from a experiment and with the Reynolds-averaged (turbulent) Navier-Stokes computation of DLR-Gottingen on a grid of about 2 million cells. The Euler results show the single suction peak corresponding to a primary separation from the leading edge. The secondary separation at x/c_{root} of 0.6 and 0.8 occurs experimentally at 81 and 83 percent semispan, respectively; the Navier-Stokes calculations predict secondary separation at 78 percent at the same longitudinal locations. The secondary separation induces a secondary suction peak outboard of the primary and is noticeably stronger in the computation than experiment. The results of the Reynolds-averaged Navier-Stokes method demonstrate an improved correlation of theory and experiment.

4.3.6 Wing Canard

The Euler equations, in accounting for the primary leading-edge vortices, can be used to determine the principal interactions between components, including the interaction of free vortices with lifting surfaces.[23,24] An example of the correlation with the finite-volume Euler TEAM code and experiment for a wing canard at transonic conditions (M_∞=0.9, α=4 deg) is shown in Fig. 4.3.9. The wing pressures are shown with and without the canard surface. The influence of the canard is to decrease the wing leading edge pressures. The pressure computations agree well with the experimental values, except in the immediate vicinity of the wing upper surface trailing edge, attributed to a local shock/boundary-layer interaction. The increment in pressures from canard off to canard on is predicted more closely than the wing pressures, as is generally expected. Scherr and Das[25] draw similar conclusions from Euler computations of a slender canard-delta configuration at high angle of attack. More demonstrations of the capabilities of Euler methods in free vortex simulations can be found in Ref. 24.

4.3.7 Asymmetric Cone Flows

The flow over a cone at high angle of attack is dominated by the vortices which arise over the leeward side of the body from boundary-layer crossflow separations. These vortices exert considerable influence on the local pressure distributions and can interact with other components downstream. Extensive experimental investigations have revealed that for subsonic crossflow conditions, the flow field remains symmetric until the value of angle of attack exceeds approximately 2–3 times that of the nose half-angle.[26] The flow then is characterized by a markedly asymmetric pattern of vortices, giving rise to large side forces and lateral instabilities. Since these vortices arise from viscous separations over a smooth surface, computational studies require direct account of viscosity (i.e., the Euler equations need to be augmented with a boundary-layer or empirical separation model).

Because of the reduced computational cost and the ability to perform parametric studies easily, considerable insight has been gained through the use of inviscid methods with prescribed separations. Fiddes[27] and Fiddes and Smith[28] assumed incompressible, small disturbance flow and modeled the vortical flow over a circular cone as either a concentrated line vortex or a vortex sheet. With prescribed symmetric separation points, two families of solutions were found at angles of attack beyond twice the cone half-angle: a symmetric solution and a pair of mirror-image asymmetric solutions. The side force variations from the asymmetric solutions were in reasonable agreement with experimental results, thus providing evidence that the origin of the asymmetry is inviscid in nature. Also, with asymmetric separation points prescribed from experiment, two stable families of asymmetries were found: one with a small side force and a slight asymmetry and the other with a large side force and a pronounced asymmetry. The larger asymmetry family produced side force values which were on the order of the side force values measured experimentally.

Marconi[29] solved the Euler equations at supersonic speeds using the conical equations and a prescribed-separation model, thus removing the small disturbance limitation of Fiddes. The results obtained were in substantial agreement with the previous findings of Fiddes, in that a pair of mirror-image asymmetric solutions were found at angles of attack greater than twice the cone half-angle. The results for a 7-deg half-angle cone at M_∞= 1.6 are shown in Fig. 4.3.10. The streamlines and surface pressures at alpha=23 deg indicate the asymmetric flow pattern; one vortex is located closer to the body and farther from the plane of symmetry than the other. The onset of asymmetry is shown versus angle of attack for 7-deg and 5-deg cones. The 5-deg cone computations were made with symmetrical separations prescribed at 120 and 150 degrees; the prescribed separation location of 150 deg shows better agreement with the experiments. Beyond the point of asymmetry, complete agreement would not be expected since viscous effects would asymmetrically change the separation locations from those prescribed. In contrast to the findings of Fiddes, Marconi found that in the range where asymmetries occurred, the symmetric solutions (obtained with symmetry imposed) were unstable and always evolved to asymmetric solutions. Thus, the Euler computations have yielded valuable insights into the nature of the asymmetric flows; further studies, including Navier-Stokes computations, are summarized in Ref. 30.

4.3.8 References

1. HUMMEL, D., "On the Vortex Formation Over a Slender Wing at Large Angles of Incidence," AGARD CP-247, Paper 15, Feb. 1983.

2. KJELGAARD, S. O., SELLERS, W. L., III and WESTON, R. P., "The Flowfield Over a 75 Degree Swept Delta Wing a 20.5 Degrees Angle of Attack," AIAA 86-1775-CP, June 1986.

3. CARCAILLET, R., MANIE, F., PAGAN, D. and SOLIGNAC, J. L., "Leading Edge Vortex Flow Over a 75 Degree-Swept Delta Wing – Experimental and Computational Results," ICAS-86-1.5.1, Sept. 1986.

4. PAYNE, F. M., NG T. T., NELSON, R. C. and SCHIFF, L. B., "Visualization and Flow Surveys of the Leading Edge Vortex Structure on Delta Wing Planforms," AIAA 86-0330, Jan. 1986.

5. STANBROOK, A. and SQUIRE, L. C., "Possible Types of Flow at Swept Leading Edges," Aeronautical Quarterly, Vol. 15, 1964, pp. 72-82.

6. SZODRUCH, J. and GANZER, U., "On the Lee-Side Flow Over Delta Wings at High Angles of Attack," Paper 21 in High Angle of Attack Aerodynamics, AGARD-CP-247, Jan. 1979.

7. WHITEHEAD, A. H., JR., HEFNER, J. N. and RAO, D. M., "Lee-Surface Vortex Effects Over Configurations in Hypersonic Flow," AIAA 72-77, Jan. 1972.

8. MILLER, D. S. and WOOD, R. M., "Lee-Side Flow Over Delta Wings at Supersonic Speeds," NASA TP-2430, June 1985.

9. MCMILLIN, S. N., THOMAS, J. L., and MURMAN, E. M., "Navier-Stokes and Euler Solutions for Lee-Side Flows Over Supersonic Delta Wings : A Correlation with Experiment," NASA TP-3035, Dec. 1990.

10. POWELL, K. G., "Vortical Solutions of the Conical Euler Equations," *Notes on Numerical Fluid Mechanics*, Vol. 28, Vieweg, 1990.

11. MURMAN, E. M., POWELL, K. G., MILLER, D. S. and WOOD, R. M., "Comparison of Computations and Experimental Data for Leading Edge Vortices - Effects of Yaw and Vortex Flaps," AIAA 86-0439, Jan. 1986.

12. MONNERIE, B. and WERLE, M., "Etude de l'Ecoulement Supersonique & Hypersonique d'une Aile Elancee en Incidence," Paper 23 in AGARD CP-30, 1968.

13. LONGO, J.M.A. and DAS., "Numerical Simulation of Vortical Flows over Close-Coupled Canard-Wing Configuration," AIAA 90-3003, 1990.

14. BRENNENSTUHL U., "Experimentelle und Theoretische Untersuchungen Uber die Wirbelbildung an Doppeldeltafluegeln," Disertation TIJ Braunschweig, Germany, 1985.

15. AGRAWAL, S., BARNETT, R. M. and ROBINSON, B. A., "Investigation of Vortex Breakdown on a Delta Wing Using Euler and Navier-Stokes Equations," MCAIR Report 90-021, McDonnell Aircraft Company, Oct. 1990.

16. HITZEL, S., "Compressible Reibungsfreie Vorderkantenwirbelstrukturen an Schlanken Flügeln — Theoretische Überlegungen und Hochauflösungssimulation," DGLR-Bericht 90-06, 1990, pp. 181-185.

17. HITZEL, S. "Low and High Speed High Angle-of-Attack Flow Around a Delta Wing by an Euler Simulation," Proc. of the Intl. Conf. on the Prediction and Exploitation of Separated Flow, Paper No. 24, Royal Aeronautical Society, London, April 18-20, 1989.

18. HITZEL, S., "Wing Vortex Flows Up Into Vortex Breakdown — A Numerical Simulation," AIAA 88-2518, 1988.

19 WILLIAMS, B. R., KORDULLA, W., BORSI, M. and HOEIJMAKERS, H. W. M., "Comparison of Solution of Various Euler Solvers and One Navier-Stokes Solver for the Flow About a Sharp-Edged Cropped Delta Wing," Paper No. 2, AGARD CP-494, Vortex Flow Aerodynamics, Oct. 1990.

20. ELSENAAR, A. and HOEIJMAKERS, H. W. M., "An Experimental Study of the Flow Over a Sharp-Edged Delta Wing at Subsonic, Transonic and Supersonic Speeds," Paper No. 15, AGARD CP-494, Vortex Flow Aerodynamics, Oct. 1990.

21. ELSENAAR, A. and ERIKSSON, G. (EDS.), "Proceedings of the Symposium on the International Vortex Flow Experiment on Euler Code Validation," FAA Publication, Stockholm, Oct. 1986.

21. ELSENAAR, A., HJELMBERG, L., BUTEFISCH, K. A. and BANNINK, W. J., "The International Vortex Flow Experiment," AGARD CP-437, 1988.

23. RAJ, P., KEEN, J. M. and SINGER, S. W., "Applications of an Euler Aerodynamic Method to Free-Vortex Flow Simulation," AIAA 88-02517, June 1988.

24. RAJ, P., "Aerodynamic Analyses Using Euler Equations. Capabilities and Limitations," Chapter 18 in *Applied Computational Aerodynamics*, AIAA Progress in Astronautics and Aeronautics, Vol 125, 1989, pp. 679-700.

25. SCHERR, S. and DAS, A., "Basic Analysis of the Flow Fields of Slender Delta Wings Using the Euler Equations," ICAS Congress, Vol. II, Jerusalem, 1988, pp. 1428-1436.

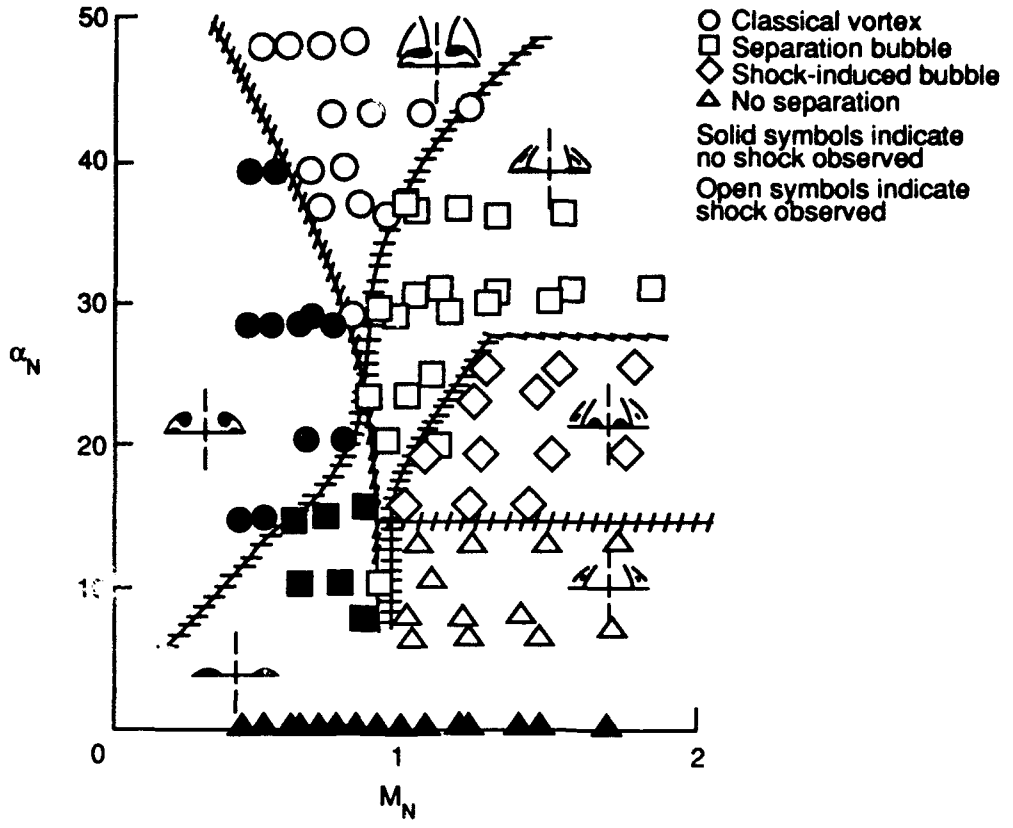
27. KEENER, E. R. and CHAPMAN, G. T., "Onset of Aerodynamic Side Forces at Zero Sideslip on Symmetric Forebodies at High Angles of Attack," AIAA 74-770, Aug. 1974.

28. FIDDES, S. P., "Separated Flows About Cones at Incidence – Theory and Experiment," *Studies of Vortex-Dominated Flows*, Springer-Verlag, New York, 1987, pp. 185-310.

26. FIDDES, S. P. and SMITH, J. H. B., "Calculations of Asymmetric Separated Flows Past Circular Cones at Large Angles of Incidence," Paper 14, AGARD CP-336, 1982.

29. MARCONI, F., "Asymmetric Separated Flows About Sharp Cones in a Supersonic Stream," *Lecture Notes in Physics*, Vol. 323, Springer-Verlag, New York, 1989, pp. 395-402.

30. THOMAS, J. L. and HARTWICH, P. M., "Navier-Stokes Analyses of Flows Over Slender Airframes," Chapter 12 in *Tactical Missile Aerodynamics : Prediction Methodology*, Vol. 142 in AIAA Progress in Astronautics and Aeronautics, pp. 561-648.



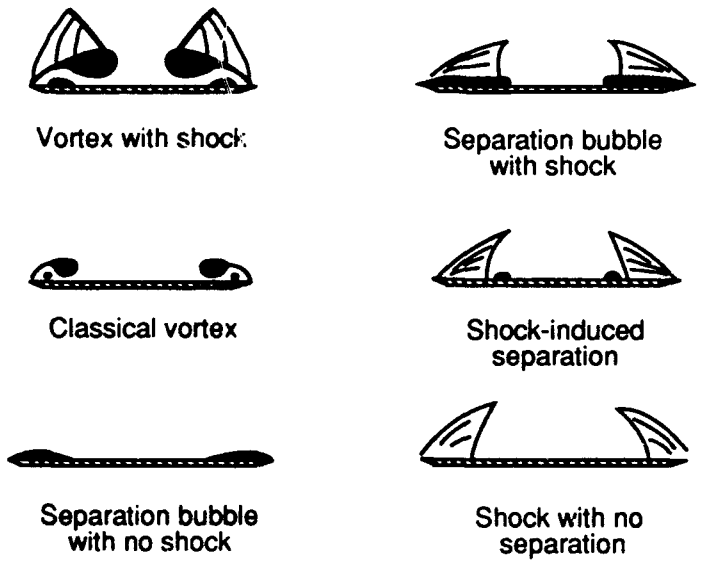


Figure 4.3.1 Classification of experimental data for supersonic flow over sharp-leading-edge delta wings.

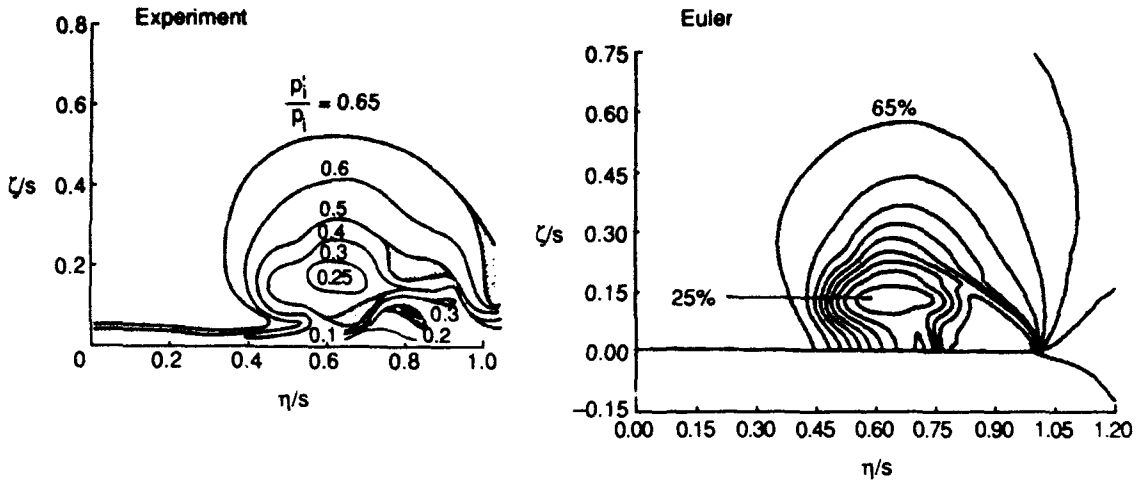


Figure 4.3.2 Comparison of measured (left) and Euler (right) pitot pressures for 75 deg swept delta wing; $M_\infty = 1.95$, $\alpha = 10$ deg.

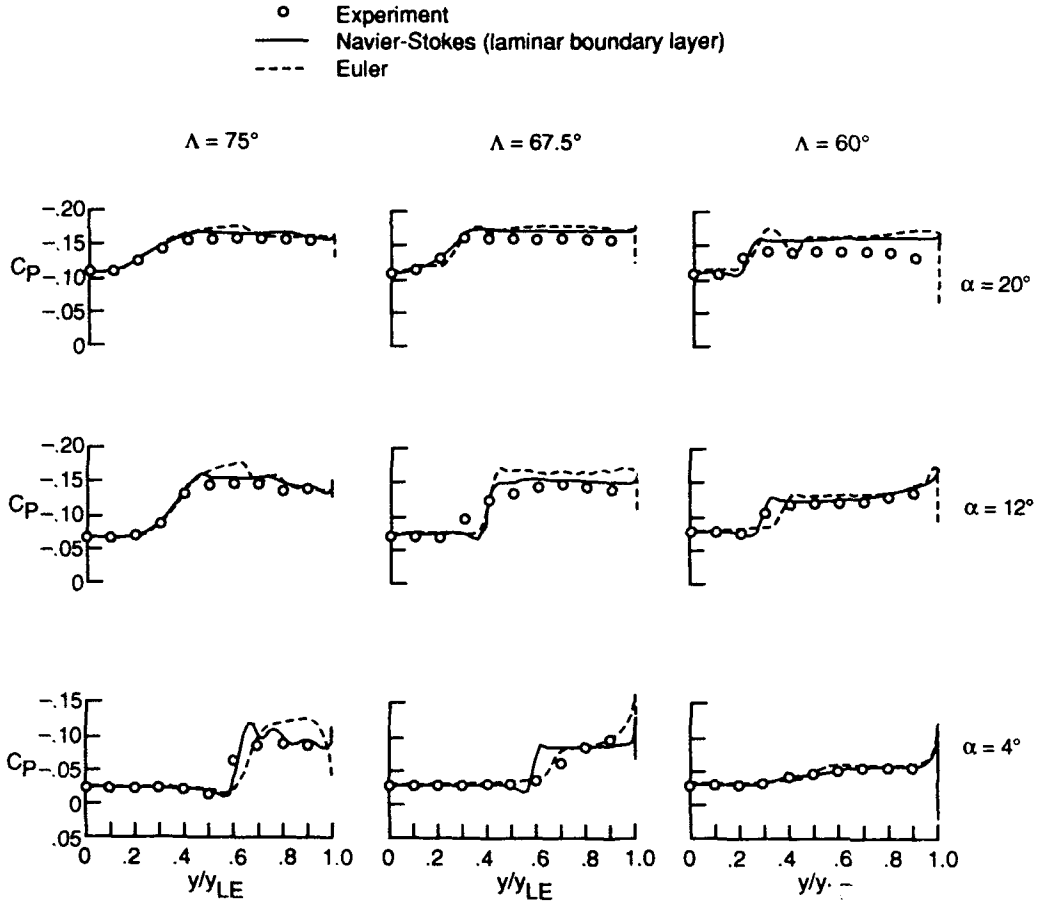


Figure 4.3.3 Comparison of experimental, conical laminar Navier-Stokes, and conical Euler computational surface pressure distributions.

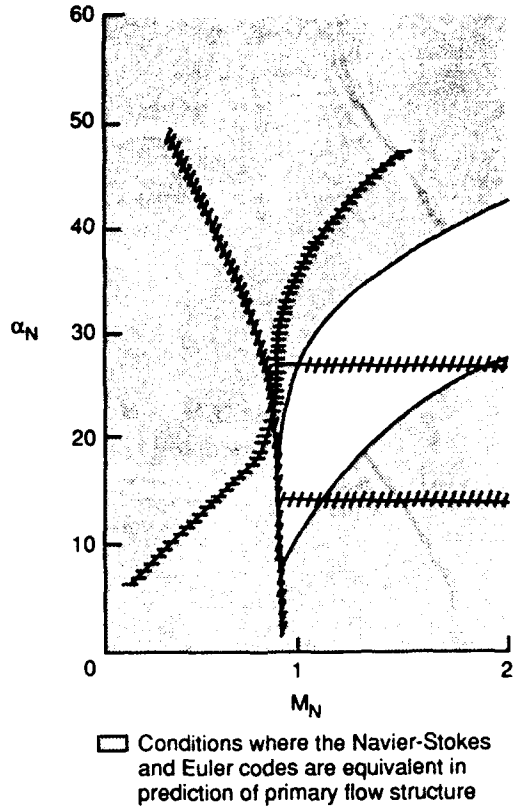


Figure 4.3.4 Envelope of conditions at which the Euler codes predict correctly the primary flow structure in supersonic wing design.

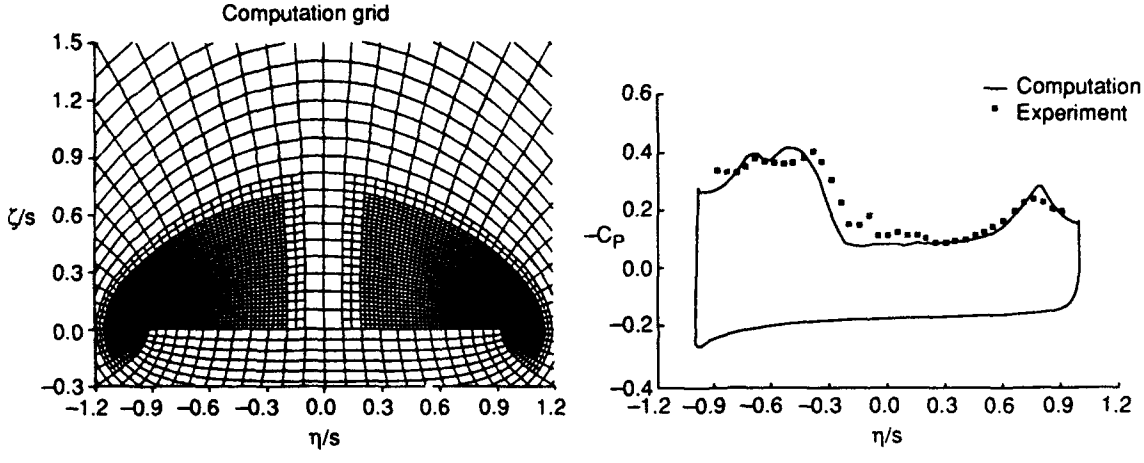


Figure 4.3.5 Comparison of adaptive conical-flow Euler with experiment for yawed 75 deg swept delta wing; M_∞ = 1.7, α = 12 deg, β = 8 deg.

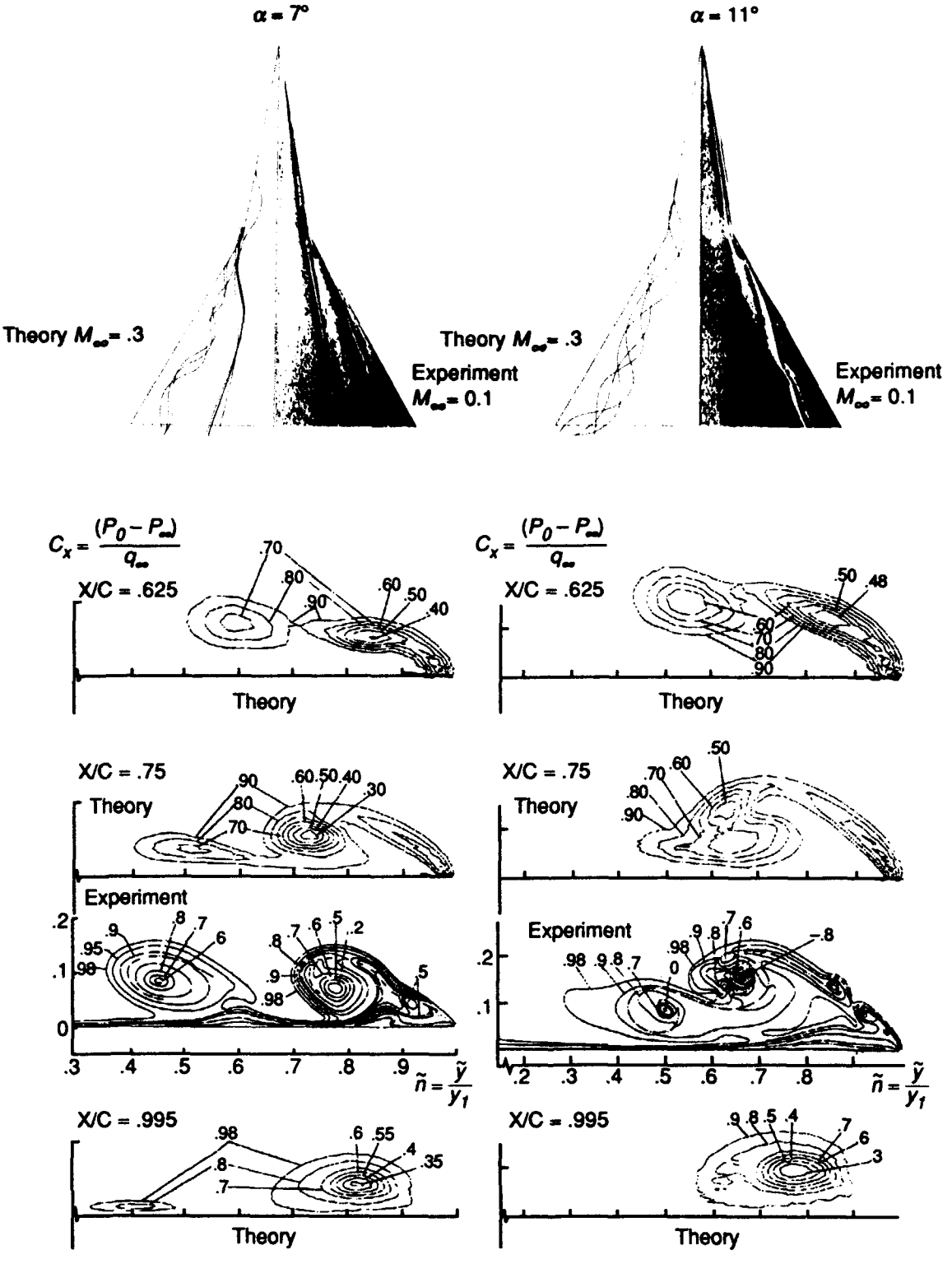


Figure 4.3.6 Euler predictions of flow over a double-delta wing at subsonic conditions.

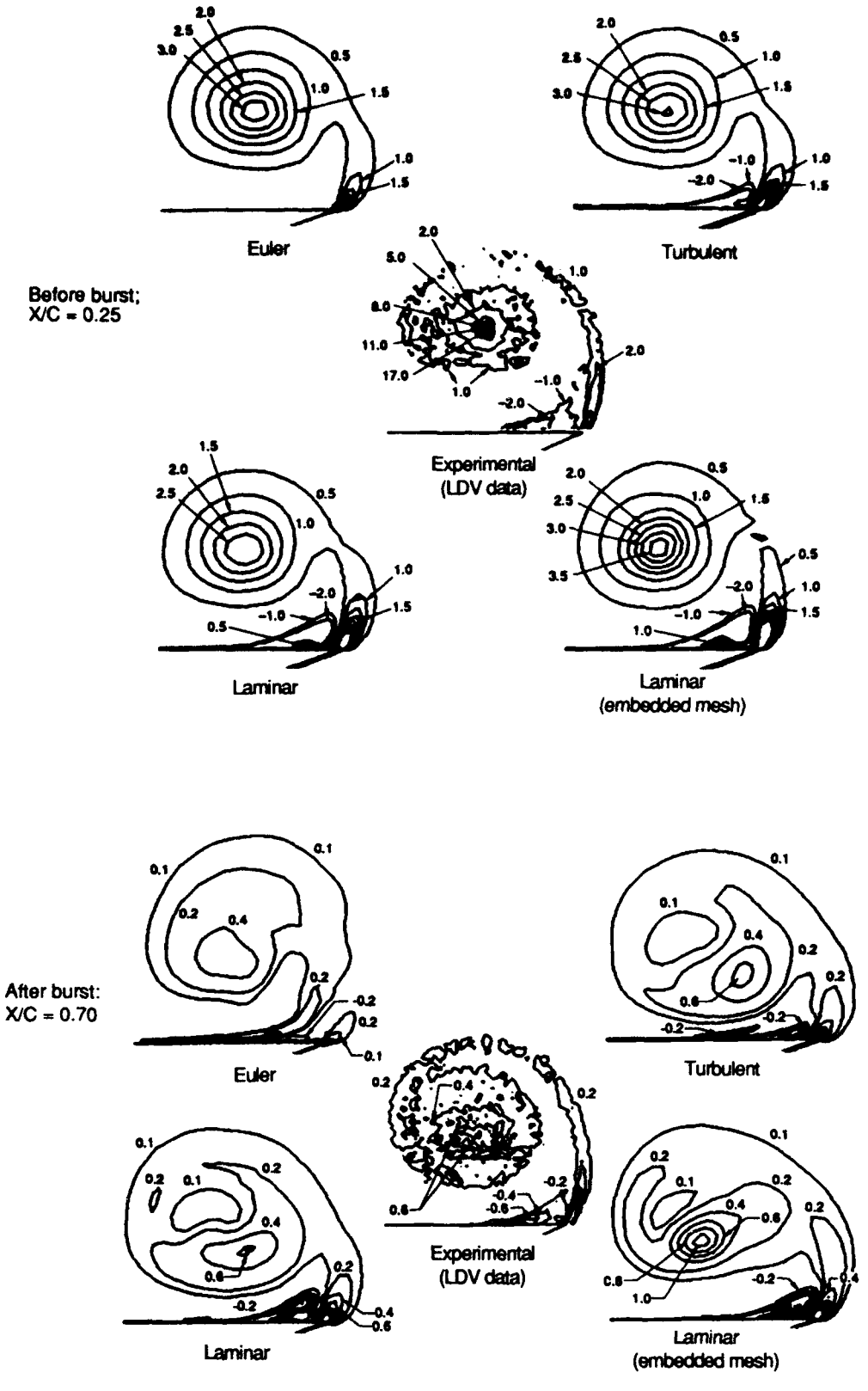
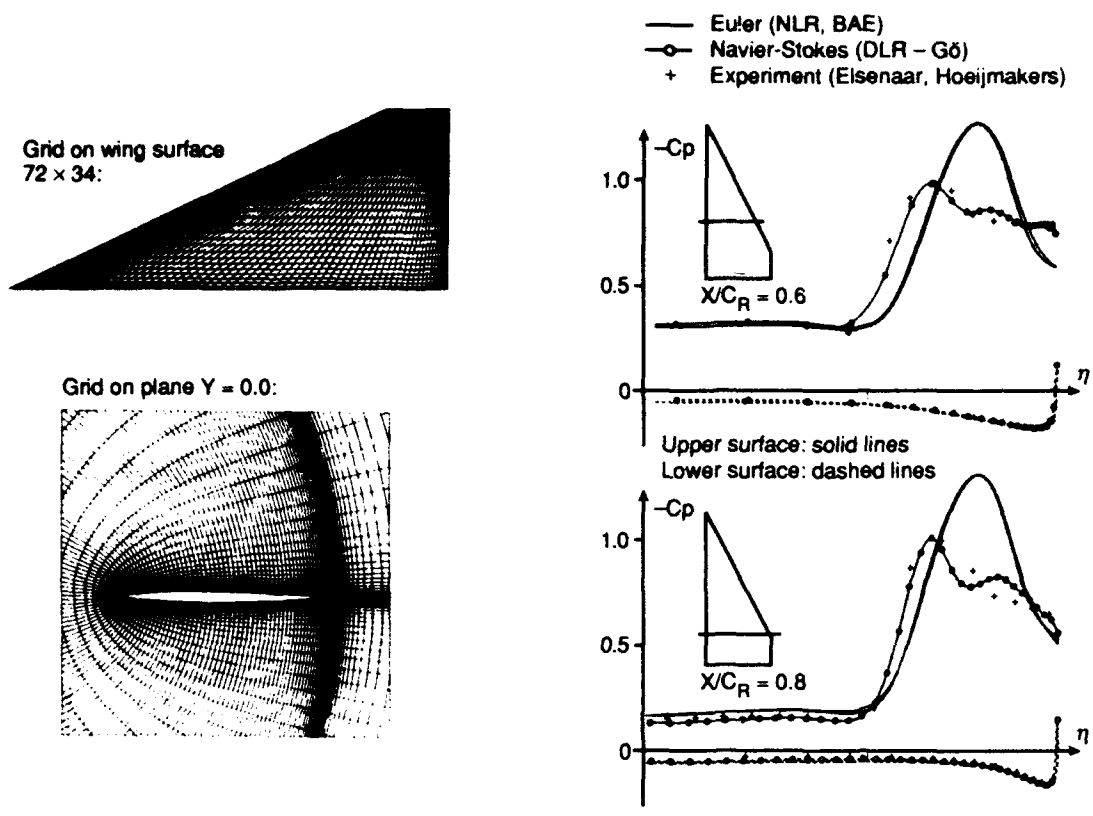
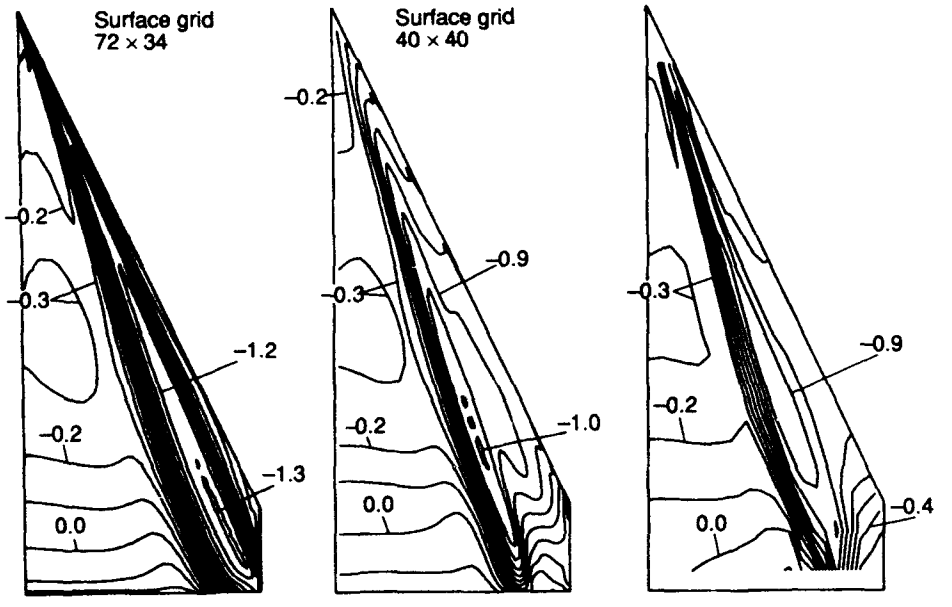


Figure 4.3.7 Comparison of predicted vorticity levels (x-direction) from Euler and Navier-Stokes solutions; M_∞ = 0.3, α = 30 deg, Re_c = 1 x 10^6.



Top view of surface pressure contours



EULER (NLR)
M_∞ = 0.85, α = 10 deg
300K grid

NAVIER STOKES (DLR – Gö)
M_∞ = 0.85, α = 10 deg, Re = 9M
2M grid

EXPERIMENT
M_∞ = 0.85, α = 10 deg, Re = 9M

Figure 4.3.8 Comparison of computational methods with experiment for the flow about a sharp-edged cropped delta wing; M_∞ = 0.85, α = 10 deg.

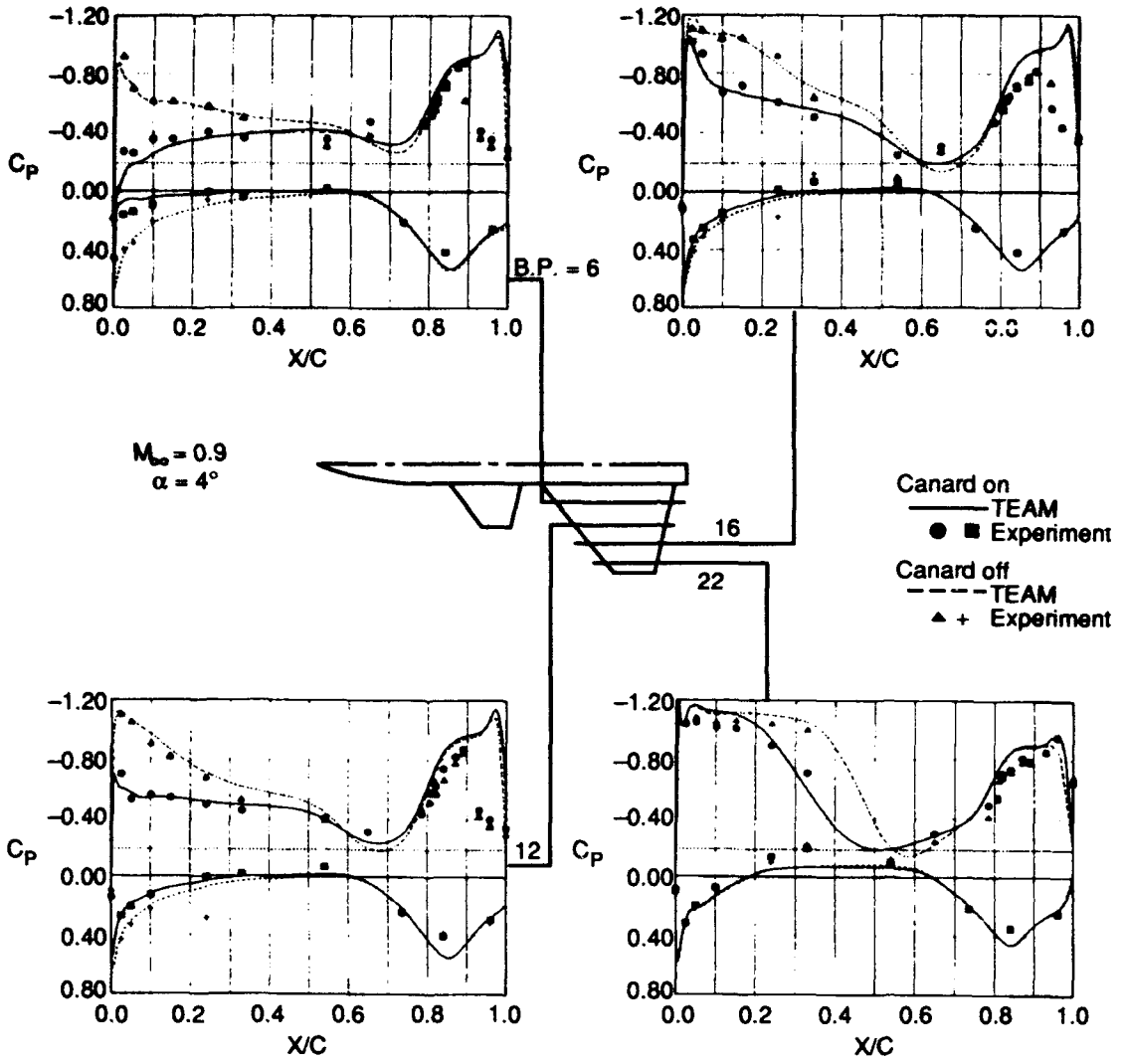
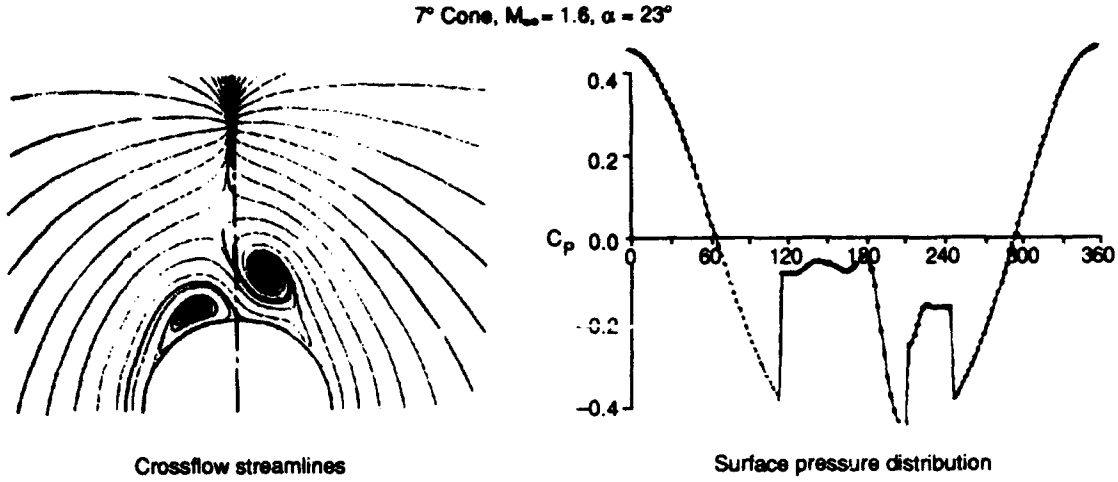


Figure 4.3.9 Correlation of computed and measured surface pressure distribution for a canard-wing-body configuration; M_∞ = 0.9, α = 4 deg.



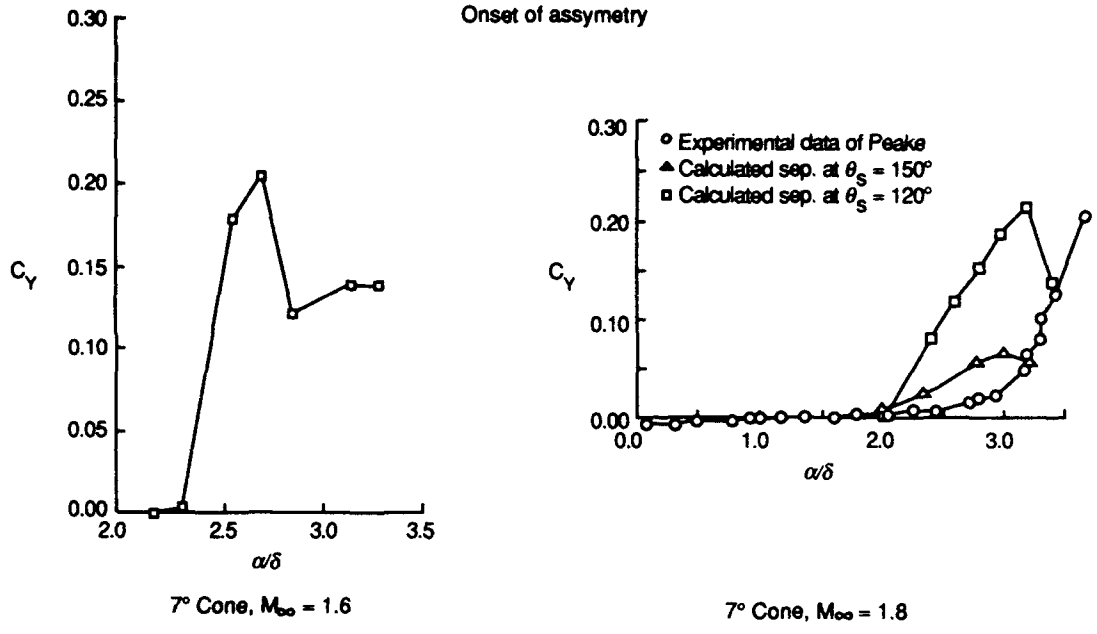


Figure 4.3.10 Asymmetric conical-flow prescribed-separation Euler computations.

4.4 AIR-VEHICLE CONFIGURATIONS

The representative capabilities of Euler solvers to predict the inviscid flow over general air-vehicle configurations are described below. The configurations able to be treated are geometrically complex, including representations of wings, bodies, pylons, nacelles, flap-track fairings, flaps, elevons, empennage, etc. The examples shown use both structured- and unstructured-grid methods, which are tailored to the application areas of interest. The use of Euler methods to treat engine-airframe integration studies is widespread, as shown for the subsonic transport examples. The capabilities of Euler solvers for fighter-type configurations are demonstrated in several applications. An example from the advanced tactical fighter (ATF) development illustrates an application of an Euler solver to an aircraft design which proved useful in complementing wind-tunnel experiments and uncovering some deficiencies early in the design cycle. An example application of the sonic boom signature for a supersonic transport aircraft is shown; the Euler equations are necessary to supplement linear theory methods as the Mach numbers approach 3. A complete cruise missile simulation with a counterrotating propfan propulsion system is shown; time-accurate simulations were used to determine model loads before construction of a large-scale powered test model. Application to the prediction of store carriage flow fields are shown with both structured- and unstructured-grid methods. For reentry configurations, the application of multiblock structured-grid Euler solvers to the prediction of flap loads for a Hermes reentry configuration is shown, as well as computations for the U.S. Space Shuttle configuration.

4.4.1 Subsonic Transport Aircraft

Structured-Grid Applications

Several examples which demonstrate the capabilities of Euler solvers to calculate transonic flows over jet transport aircraft are shown below. At Deutsche Airbus GmbH (DA),[1] the multiblock multigrid Euler integration algorithm MELINA, together with the INteractive GRID generation system INGRID and the Practical Interactive Solution Analysis system PISA for post processing, forms the tool package for three-dimensional inviscid compressible flow analysis. Problems of complex geometry (body/wing/pylon/engine) and of jet or nacelle flow can be solved. The Euler code MELINA is a Jameson-type, explicit, multiblock, multigrid, cell-vertex code and is continuously upgraded and adapted for the applications which are of interest to the transport aircraft designers at DA.

With the interactive algebraic grid generator INGRID, several tasks can be tackled. It serves as a geometry definition and manipulating system to create the configuration to be evaluated from given input data. At any stage of the construction, user-defined curves describing the geometry can be graphically displayed for error checking and judgment of the configuration. In a second step, the surface is covered with a surface grid with full user control of the node distribution. Then a global multiblock mesh can be generated for a wing-body configuration. In order to add further components like pylon, engine, tail etc., local blocks are then cut out of the global mesh and reconfigured corresponding to the components.

The CFD system described above is used alongside other applications to analyze the problems arising with engine/airframe integration. Two different calculations are compared with each other and with wind tunnel measurements: a body/wing configuration and a body/wing/pylon/nacelle configuration of a modern transport aircraft.

Plate 4.1 shows the general arrangement of the configuration with pylon and nacelle. The distribution of the pressure coefficient is projected onto the surface. The spanwise stations, at which experimental pressure data near the pylon were available, are depicted in Fig. 4.4.1 together with the surface grid on the lower wing side. Comparisons of the computed pressure distributions for the configuration with and without pylon and nacelle and the corresponding experimental data show a good correspondence in all cases, considering that the code is inviscid. The flow at the leading edge is modelled with high accuracy, which is a consequence of the correct angle of attack in the Euler calculation. The wing was decambered to account for boundary layer effects; thus, there was no need to modify the experimental angle of attack for the inviscid flow calculation. The residual discrepancies between experiment and simulation occur in the region behind the shock, at the lower wing side, and at the trailing edge. The problems at the shock (shock/boundary-layer interaction) and the trailing edge are a consequence of lacking boundary layer thickness modeling in the code. Some differences at the lower wing side mainly result from flap track fairings that were mounted on the wing in the experiment and were not modelled in the simulation.

In order to find out whether the engine installation effects are predicted accurately with the Euler code MELINA, the pressure distribution of the engine mounted configuration and the clean wing case were subtracted from each other (Fig. 4.4.1). It was expected that due to the subtraction the uncertainties, inherent in both the method (simplified viscous effects) and the experiment could be eliminated and that the effect of the flap track fairings which were not present in the numerical simulation could be filtered out. Positive and negative Cp values correspond to deceleration and acceleration of the flow due to the engine installation, respectively. There is an excellent correlation between the interference effects of theory and experiment.

In view of the development of ultra-high bypass (UHB) engines, the aerodynamic interference between airframe and engine becomes more and more important. Figure 4.4.2 shows the capability of Euler solvers to simulate the flow field around the DLR-ALVAST wing-body combination[2] with different wing-mounted engines. The CFM-56 engine represents the conventional engine and the UHB engine corresponds to the DLR-CRUF simulator.[3] The geometry of the wing-body combination represents a typical modern wide-body transport aircraft of Airbus type. The flow field computations for the different configurations have been performed by the use of the DLR Euler-code CEVCATS. This code is written in a block-structured form using a multigrid acceleration technique and allows an arbitrary application of boundary conditions on the block faces.[4]

The first step in the multiblock approach for complex configurations is to decide on a global grid topology. In case of a wing-body-engine-pylon (wbep) configuration, an H-type structure in the streamwise direction is used; an O-type structure is used in the spanwise direction for the wing-body grid and in the circumferential direction for the engine (Fig. 4.4.2).

A three-dimensional view of the field grids for the different engine types is also shown. Both field grids consist of 11 computational blocks with a total of approximately 600,000 mesh cells.

Transonic pressure distributions (M_∞ = 0.75 and α = 0.84 deg) at different spanwise sections for both engine types are compared to those of the wing-body configuration. The presence of the engine results in a forward movement of the shock and influences the complete upper surface of the wing. On the lower side of the wing, the influence is more local and the closer position and the larger geometry of the UHB engine causes a stronger interference. Due to the movement of the shock caused by the propulsion system, lift is considerably decreased, as shown in the lift versus semispan comparisons. In case of the UHB engine, an additional loss of lift occurs in the region around the engine location.

Computational time for a single flow calculation is about half an hour on a CRAY-YMP. Comparisons between calculated and measured pressure distributions demonstrate the capability of the CEVCATS code to predict essential interference effects due to the propulsion system.[5,6] The capabilities shown above are typical of the geometric complexities which can be analyzed with multiblock structured-grid codes.[7-10]

Unstructured-Grid Applications

The unstructured-grid methods have also been applied to complex transport configurations, as shown below for a generic McDonnell Douglas Corporation (MDC) Tri-Jet configuration.[11] Generally, in this latter approach, considerably fewer manpower resources are expended to generate the field grids. The surface triangulation and the pressure contours of the under-wing engine at transonic conditions of M_∞ = 0.825 and α = 1.0 deg are shown in Fig. 4.4.3. The conditions correspond to unforced flow through the engine-cowl components. The grid and solutions were computed with the AIRPLANE program developed by Jameson and Baker.[12] The surface definition required two days; the flow field mesh contains 384,914 nodes and over 2 million tetrahedra. The grid is determined by triangulating a series of graded-refinement Cartesian meshes; the flow solver advances in time using Runge-Kutta time stepping in combination with a central-differenced residual with explicitly added dissipation terms.

The pressure distributions on the wing at 24 and 32 percent semispan show good agreement with experimental measurements. The pressure peaks and general behavior are predicted well with a slight discrepancy near the shock on the wing upper surface. The capability to study component interference is also shown through wing pressures with and without the under-wing engine. On the lower wing, the retardation effect ahead of the pylon and the acceleration over the aft end closely matches the experimental trend. In addition, the changes in pressures due to the nacelle and pylon compare well. In this case, less than one man-minute of editing was required to remove the engine assembly from the configuration; the resulting field-grid generation required 18 minutes on a single-processor CRAY-YMP supercomputer.

The transonic flow over a transport configuration computed with an unstructured-grid Euler method[13] is shown in Figs. 4.4.4–4.4.5. The unstructured surface-grid modeling is detailed and includes the wing-body-pylon-nacelle geometry as well as the coverings of the flap-deployment mechanisms at four semi-span positions on the wing. The engine is modeled as an actuator disk: inflow and outflow conditions are specified at upstream and downstream cross-sectional stations. The pressure contours over the configuration are shown at a transonic cruise condition: M_∞ = 0.801 and α = 2.738 deg; the longitudinal pressure variations at a circumferential position located 22 deg outboard from the centerline of the nacelle shows excellent agreement with experimental data. On the wing, the Euler computations at the semispan location of 0.293 indicate an upper surface shock located further aft than the experimental results, as expected because of boundary-layer interaction effects. On the subcritical lower surface, the chordwise variations due to the presence of the pylon-nacelle and the streamwise flap-track fairings are substantial and are well predicted by the inviscid method.

A similar capability is shown in Plate 4.2 for a Dassault Aviation transport configuration[14,15] with fuselage-mounted nacelles and vertical and horizontal control surfaces. The Mach contours are shown at a transonic condition[15] (M_∞ = 0.85, α = 3.0 deg). The surface triangulation shown is the result of two successive refinements and has 40,000 nodes. The numerical formulation uses upwind approximations based on Osher's approximate Riemann solver and MUSCL interpolation. The time advancement is implicit and is combined with adaptive mesh refinement and unnested multigrid acceleration techniques. The solutions were obtained after a three-order-of-magnitude reduction in the steady-state residual equations, corresponding to 300 iterations at a Courant number of 20.

4.4.2 Supersonic Transport Aircraft

Recently, new initiatives towards the design of a follow-on to the Concorde supersonic transport have begun. Studies have focused in the area of configuration design for sonic boom minimization in order to mitigate the noise associated with the sonic boom and determine the feasibility of overland supersonic flight. The sonic boom signature for two supersonic transport configurations was studied by Siclari and Darden.[16] A central-difference finite volume method in the cross-flow planes and an implicit upwind finite difference technique in the marching direction was used to solve the three-dimensional unsteady Euler equations.

Computer codes used in the design and analysis of low boom configurations have traditionally been based on Whitham's modified linear theory analysis,[17] which was extended to apply to lifting bodies by Walkden.[18] Studies have demonstrated that the traditional modified linear theory methods become inaccurate as the free stream Mach number approaches 3. At higher Mach numbers, stronger shocks are generated and the assumption of isentropic flow becomes invalid. Thus, Euler and Navier-Stokes methods applicable to the area of sonic boom prediction and minimization at Mach numbers above 2.7 are needed. These near-field prediction methods can provide detailed flow field information for guidance in component integration or provide flow field input for nonlinear or modified linear theory extrapolation methods.

Two low boom aircraft concepts,[19] designed for low sonic boom at Mach 2 and Mach 3, which were designed, built, and tested at NASA, are schematically shown in Fig. 4.4.6. The Mach 2 configuration has a flat platypus nose and the Mach

3 concept has a needle nose. Both have highly swept leading edges inboard with cranked lower sweep outboard wings for improved low-speed performance. Both concepts were studied without engine nacelles; the corresponding wind-tunnel data were also available for comparison.

Approximately 850,000 points were used to compute the flow in the vicinity of each aircraft, and approximately 1.1 million points were used to compute the flow to 15 body lengths downstream of the aircraft. The computational surface model and computed aft pressure contours at M_∞ = 3.0 and α = 1.97 deg for the Mach 3 low boom configuration are shown in Fig. 4.4.6. The cranked wing generates a strong shock as indicated by the isobars. The computed isobars in a plane at the end of the sting show a strong shock in the leeward region, attributed to the coalescence of the wing trailing-edge shock and sting attachment shock. On the windward side, a strong shock occurs due to the wing crank; the circular isobars just to the right and left to the sting are vortices generated by the wing tips expansions.

The computed pressure signatures are compared to recent wind tunnel model data for both the Mach 2 and Mach 3 configurations. Both models were 1/300 scale or about 12 inches in length. The wind tunnel model data were converted to full scale in feet to compare to the computations. The wind tunnel data were taken at two different distances below the aircraft for each configuration. For both configurations, good correlation with the data is shown for both distances for the forward half of the signature. At h/ℓ = 0.5, the Mach 2 data show a series of shocks and expansions in the last half of the signature. The computation shows a single shock and expansion. At h/ℓ = 1.0, slightly better correlation is achieved. The data stills show a series of shocks and expansions with a very large, final, expansion twice that of the computation. Virtually the same type of correlation is shown for the Mach 3 configuration. Further studies[16] indicate that it is necessary to consider three-dimensional effects in the design of low boom concepts, since the Mach 2 concept showed sonic booms along the side of the ground footprint with magnitudes as much as 40 percent greater than those directly along the flight path axis.

4.4.3 Fighter Aircraft Configurations

In 1986, Eberle and Misegades[20] presented some of the first inviscid solutions of a complete fighter aircraft using a high-resolution Euler code. The underlying numerical method uses a Godunov-type averaging procedure based on the eigenvalue analysis of the Euler equations; the fluxes are evaluated at the finite volume faces, thus generating separate constant sets of flow variables on either side. The procedure is third-order accurate on equidistant meshes (in one dimension) and locally monotonicity preserving, which seems to avoid the drawbacks of global TVD schemes.[21]

The grid generation for complex configurations like a fighter aircraft is performed from solutions of linear biharmonic equations in which only one parameter is necessary to be prescribed. H-type grids are used in a monoblock approach where specific coordinate planes are made coincident with certain surface elements of the configuration at hand. With this procedure, dummy grid points are generated inside the configurations which have to be blanked out. The resulting code can be run for steady state solutions by using either an explicit time stepping integration scheme or a point-implicit relaxation scheme; the latter scheme is implemented as a point Jacobi extrapolation procedure.

The aircraft model studied was based on design studies for the early MBB-ACA fighter conducted in 1985. The computations were carried out on a CRAY-XMP/2 computer using a mesh with a total of 520,000 grid points for modelling the complete configuration.

Plate 4.3 shows the pressure distribution on the aircraft for the following flow conditions: M_∞ = 0.85, α = 7.5 deg, β = 5 deg. To enable a realistic simulation, mass flow through engine intakes and nozzles was allowed based on the following parameters: p_{jet}/p_{intake} = 1, m_{jet}/m_{intake} = 3, M_{intake} = 0.75. The windward side (due to yaw) is the left side of the aircraft. Shocks can be detected at the canard and the wing trailing edges. Plate 4.4 shows temperature contours over the aircraft and in the region downstream of the engine exhaust at transonic conditions. A good estimate of the accuracy achieved with the Euler solver applied to a complete airplane can be obtained by checking the total pressure distribution. Although not shown here, the windward surfaces of the configuration do not exhibit total pressure variations except at part of the fuselage underside; the errors in this region were traced to a triangular interpolation between input points which was too disparate.

F–18 Aircraft

A demonstration of the complete vehicle modeling capability available with unstructured-grid methods[22] is shown in Fig. 4.4.7. The surface geometry of a complete F-18 fighter configuration is described by 37 surface components and 87 line components. The surface mesh extends into the engine inlet and exhaust and the half-domain discretization consisted of nearly 500,000 tetrahedral elements and 100,000 nodal points. The inlet conditions took the form of a specified Mach number of 0.4 and the exit conditions specified a jet pressure ratio of 3. The pressure distribution at a transonic Mach number of 0.9 and 3 deg angle of attack is shown. The results shown were obtained in 1990[22] and were initial demonstrations of the power of the unstructured-grid methods for aerodynamic applications; the entire time from surface geometry demonstration to flow solution was less than two weeks.

Advanced Fighter Configuration

Another example of the unstructured-grid Euler capability for complex configurations is the application to an advanced fighter configuration developed by Alenia Aeronautica, shown in Fig. 4.4.8. The results are from a validation study to assess the capability of Euler methods, with emphasis on the transonic regime, for a new generation of fighter aircraft.[23] The grid was generated using the advancing-front grid generation of Peraire et al.[24], which allows highly anisotropic meshes (i.e., meshes where the elements can be stretched along arbitrarily-oriented directions) and is tightly coupled to an existing CAD system. The geometric modeling is detailed, includes the inlet, exhaust, control surfaces, flaptrack fairings, store pylons, and tip pods. The definition of the surface consists of 154 CAD surfaces and the construction of the surface mesh required one week. The generation of the volume mesh was done overnight using an engineering workstation. The grid consisted of 141,339 nodes and 763,566 tetrahedrons.

The pressure contours at a transonic speed in slight yaw (M_∞ = 0.80, α = 4.0 deg, β = 2 deg) were computed with the UES3D code, which is an explicit node-centered finite-volume-based central-difference spatial discretization. The flux balance is done by a summation over the faces of the tetrahedrons in the volume, with each face being by two tetrahedrons. The engine simulation imposed a given mass-flow rate. The scheme requires 45 microseconds per grid point per iteration on a CRAY-YMP-2E single-processor computer and 70 words of storage per node. Solutions can be obtained in an hour of CPU time. A modification of the geometry and the corresponding flow field can be obtained in less than a day, indicating the numerical strategy can be used profitably in the industrial environment. Although not shown here, comparisons of the computations with detailed pressure measurements made during wind-tunnel tests have been reported and have shown good accordance.[23]

Advanced Tactical Fighter (ATF) Aircraft

Application of the multiblock structured-grid TEAM code to the complete advanced tactical fighter prototype configuration (for which surface pressure contours at transonic speeds are shown in Plate 4.5) began in earnest in 1988.[25-27] This aircraft features multiple lifting surfaces, twin tails, and an integrated airframe/propulsion system. Based on the capabilities of the code at the time and the schedule constraints of the project, only an inviscid Euler analysis of this complex configuration was considered feasible in order to have an impact on the design. Solutions were obtained for several transonic and supersonic Mach numbers and angles of attack for both flow-through and powered nacelles under symmetrical and asymmetrical flight conditions. Powered conditions were simulated by prescribing mass-flow ratio values at the nacelle face. The TEAM code[25] was used in a truly predictive mode for a new design. The computational solutions were obtained before the wind-tunnel pressure-model data were available. No attempt could, therefore, be made to adjust the grid or the code to improve correlations with data. Many valuable lessons learned from this exercise are summarized in Refs. 26–27.

Generating a grid on this configuration was a tedious and very time-consuming task. A few hundred man-hours and several weeks were expended to construct a 43–zone H-H type grid with about 1.5 million grid points for half the configuration. Such slow turnaround was a major impediment in evaluating the impact of numerous geometric changes on aerodynamic performance. TEAM's inability to accurately predict absolute drag levels was another disappointing aspect for the project personnel. On the other hand, detailed surface pressures predicted by the code proved valuable in estimating airloads for the structural analysis of the vehicle as well as for the thermodynamic analysis. Until data from a wind-tunnel pressure-model test became available, this capability was particularly helpful in expediting structural analysis using more realistic transonic and supersonic flow data than could be obtained using potential flow methods alone. TEAM's application to the ATF configuration was quite helpful to its developers also since it uncovered early in the design cycle some deficiencies which have since been rectified.

In Fig. 4.4.9, TEAM Euler predictions of lift coefficient at a transonic Mach number and three angles of attack are correlated with the wind-tunnel data. (The vertical axis is deliberately left blank). Good agreement is clearly seen. Correlations of computed and measured surface pressures at two stations on the wing for one condition (one close to mid-span and the other close to the tip) are also shown. In examining these correlations, caution must be exercised because: (1) the computed solutions, being inviscid, do not account for viscous effects,m and (2) even minor geometric differences between the computational and wind-tunnel models can produce relatively large changes at transonic Mach numbers. For the present case, no special effort was made to minimize any surface profile mismatch between the computational and wind-tunnel models due to time and resource constraints. The data were not collected for TEAM validation but to meet the project needs. The predicted loads were more consistent with high Reynolds number limit solutions (since boundary-layer effects are not modeled), although vortical flows from sharp leading edges are predicted. The Reynolds number for the wind-tunnel test was typically an order of magnitude less than its value in flight; thus, differences between wind-tunnel and flight pressures would be expected in regions when the flow exhibits shock-induced separation. The Euler analyses proved quite useful as one of the tools to provide information regarding configuration modifications during the evolution of the design. A summary of the CFD usage in the F-22 development program has been compiled by Bangert et al.;[27] the complete airloads prediction, including control surface variations over a range of Mach numbers, angles of attack, and sideslip was done using the TEAM Euler code and consumed 4.5 months of dedicated supercomputer usage, corresponding to 1600 CPU hours.

4.4.4 Missile Configurations

Cruise Missile

The solution depicted in Plates 4.6 and 4.7 is a single time frame (snapshot) of the unsteady pressure field around a propfan cruise missile. The geometry is essentially a complete cruise missile (as modeled in the wind tunnel) with a counterrotating propfan propulsion system operating at cruise conditions of M_∞ = 0.7 and four degrees angle-of-attack. Researchers at Mississippi State University were contracted to simulate the unsteady flow field of several potential design concepts prior to the wind tunnel testing to help estimate the material strength of the wind tunnel model to maintain structural integrity. The project, a formal cooperative effort between the U.S. Department of Defence (Navy and Air Force) and the NASA Ames and NASA Lewis Research Centers, is sponsored by the Cruise Missile Project in Washington, D.C. Its purpose is to help determine the applicability of propfan propulsion for stand-off weapons using advanced unsteady aerodynamic codes[28-35] and a large-scale power-model wind tunnel test.

A complete animation of this configuration (with the blades moving) can be formed from a collection of flow-field snapshots (a total of ninety-six for this particular configuration). All ninety-six solutions were used to predict the unsteady forces and moments on all the appendages (wings, fins, and blades) prior to the wind tunnel test. There was concern about the placement of the fins, so a fins-forward (of the propfan) configuration and a fins-aft configuration were analyzed using CFD prior to the wind tunnel experiment. CFD simulations were not limited to nondeflected fins, as in this case, but included cases involving fin deflections of five degrees. The analyses

gave an indication of the material strength necessary to withstand the unsteady aerodynamic loading on the wind tunnel model. The flow-field analysis required 4 million words of internal memory and 32 million words of solid-state storage device (SSD) on a CRAY-XMP, consuming nearly eighteen hours of CPU time to produce the solution shown. Presently, the CFD solutions and the data collected from the wind tunnel (NASA Ames) are undergoing comparative scrutiny.

ANS Missile

Applications of the Euler equations to an air-breathing missile project (ANS) studied by Aerospatiale and MBB is shown in Plate 4.8.[36] The configuration is powered by a ramjet propulsion with 4 circular air intakes and is a follow-on to the EXOCET family of missiles. The surface grid is composed of about 16,000 nodes for half of the configuration and uses a CAD surface definition. The surface pressure coefficients shown correspond to a supersonic condition (M_∞ = 2.0, α= 4.0 deg, β = 0 deg), with open air intakes operating in the supercritical regime. The computations were made with the FLU3C code, and are representative of the capability at Aerospatiale to calculate missile aerodynamics in an industrial environment. The methodology tightly couples the CAD surface generation and the mesh generation schemes and uses two numerical procedures: a space-marching procedure for supersonic flows and a multizonal approach for transonic and subsonic flows. Applications to several missiles, an Ariane 5 plus Navelle Hermes configuration, and a supersonic transport are shown elsewhere.[36]

4.4.5 Store Configurations

A good understanding of the fluid mechanics associated with carriage and release of stores from an aircraft is of primary importance to the aircraft designer. The flow field encountered on stores in the presence of an airframe is usually complex because of the many aerodynamic interactions which occur. Traditionally, the designers have relied on extensive experimental wind-tunnel tests to estimate interference effects. Recent progress in unstructured-grid methods is beginning to have an impact on that progress. For example, an extensive wing-pylon-store computation is reported in Ref. 37. The methodology consisted of an advancing-front grid generation scheme closely coupled to an upwind, finite-volume scheme. Systematic comparisons of the interference effects were made with experimental data[38] for the baseline (instore) position and two other store positions at Mach numbers of 0.95 and 1.2. The surface geometry is shown in Fig. 4.4.10 and the longitudinal variations of pressures for three store positions at M_∞ = 0.95 are compared with experiment in Fig. 4.4.11. There is a substantial effect of the store position on the pressures and the comparison with experiment is excellent at all three positions. The comparisons presented in Fig. 4.4.11 are typical of the others presented in Ref 37. The demonstrated accuracy and the fast grid generation makes the approach attractive as a preliminary design/analysis tool.

At MBB, monoblock structured-grid codes have been used since the early beginnings of field method applications. Now they have reached a high level of maturity and universality. Configurations of very high complexity can be efficiently modelled; CAD-tools like CATIA and DOGRID-5.3[39] are used as preprocessors. The resulting monoblock-structures are used in different flow solvers, but especially for Euler (EUFLEX, INFLEX) and for Navier-Stokes (NSFLEX) computations. A new technique based on smart-cell structures in monoblocks recently led to two efficient applications:[40]

- multiple overlapping monoblocks (MOM)
- dynamically overlapping grid (DOG)

MOM is an EUFLEX-type solver and is applied for steady and rigid multiple body configurations. DOG is a combination between a INFLEX-type time accurate algorithm with the SSP code[41] for the adequate time accurate representation of flight mechanically described motions and simultaneously occurring commanded control deflections.

In the present stage, both codes are able to handle two overlapping monoblock structures. The basic approach to describe the exchange of flow informations between both flow regions have been derived from descriptions of the Chimera method[42] whereby the synchronization of time between the blocks is regulated similarly to the ESE technique.[43] MOM and DOG are highly suitable for store integration investigations. As seen in Fig 4.4.12, two typical trajectory positions are shown for a missile separating from an aircraft, which has not been fully represented geometrically due to the supersonic flight conditions.

The accuracy of such predictions has already been successfully demonstrated for subsonic and supersonic cases for a variety of complicated store geometries. A good correlation between flight test (FT), wind tunnel (WT), and MOM results for a store-wing configuration has been obtained; isobars are shown in Plate 4.9 for M_∞ = 0.9. The correlated rolling moments, well-known as most sensitive to misrepresentation, agree closely, as shown in Fig. 4.4.13. These results have been achieved with only 60,000 cells in both monoblocks together.

The application of such an approach is very universal. The present code, uniquely designed for store integration studies can be easily extended to other completely different problems, even such as to the flow around a complete helicopter in forward flight inclusive ground effects, for high speed vehicle flow with realistic ground/wall effects and other nontrivial boundaries.

4.4.6 Reentry Configurations

HERMES

Computations of Rieger et al.[44] are shown for the HERMES 1.0 configuration in Plates 4.10 and 4.11 and in Fig. 4.4.14. Because of the size and specific position of controls on the configuration, the performance of the control system is strongly influenced by flow field effects through large gaps and slits between body flap, elevons, and winglet flaps. This is important insofar as the size of the controls are by far not small compared to the overall size of the vehicle. For purpose of inviscid flow field simulations, a mesh system was constructed by use of the interactive mesh generation system DOGRID developed at Dornier which consists of 7 mesh blocks and in the finest mesh possesses some 1.5 million grid points. The various mesh blocks are interconnected by use of segmentation techniques which allows the arbitrary connection of block faces or parts thereof. Particle traces are shown in Plate 4.10; Plate

4.11 displays some flow features in form of Mach number distributions on the body surface, in the symmetry plane, and in a specific cross-section. The interesting point is the behavior of the flow jet coming up on the leeside wing through the gap between elevon and winglet flap. Because of high velocities and low pressure level of the wing leeside flow, the gap jet with high pressure acts approximately like a rigid body, which produces an additional shock system that is indicated by the blue spot (low Mach number) in the fuselage end section. In the body flap end section, the interaction between the expansions around the trailing edges of the windward flow and the cross-flow phenomena, including the gap shock system, evolves to a highly complicated flow structure. The computations were performed with the block-structured code supporting a regular data structure.[44]

Reference 45 describes the application of the DLR code CEVCATS with central discretization and the upwind scheme used by ONERA's code FLU3C in order to validate inviscid flow calculations for reentry vehicles at supersonic conditions. The coordinate mesh used for this study is displayed in Fig. 4.4.14. Pressure contours in Fig. 4.4.14 show that the flow fields predicted by both codes are almost identical. The only difference in the results is a slightly better shock resolution by the upwind scheme. Detailed comparisons of computed global forces and moments with wind-tunnel measurements indicate that lift, drag, and pitching moment are in very good agreement. The differences for roll and yaw moments are somewhat larger.[45,46] Nevertheless, the effect of Mach number and angle of attack on lateral and directional stability is well predicted. Other computations for HERMES-like configurations have been made by Pfitzner[47] and Menne and Wieland;[48] a general survey of industrial applications to reentry and hypersonic configurations is given by Rieger, Stock, and Wagner.[44]

Space Shuttle

The numerical computation of an inviscid flow field about a very complex airplane and reentry spacecraft is of considerable interest to the research and designer. A unified approach to efficiently solve the Euler equations for the entire Mach number range has been developed (Euler Marching Technique for Accurate Computations (EMTAC) code) by Chakravarthy and Szema.[49,50] The approach is based on the unsteady Euler equations; an infinitely large time step and a space-marching technique is used in the supersonic flow region. The large time step makes the transient terms of the discretized unsteady equation vanish. In the subsonic flow region, a finite time step and a relaxation method are used and the steady state is approached asymptotically. A finite volume implementation of high accuracy (up to third order in one dimension) TVD discretizations is used, and thus the method is more accurate and reliable than other Euler space- and time-marching techniques based on central difference approximations. More recently, a multi-zone version of the code has been developed (EMTAC-MZ) as a superset to the single zone EMTAC. In the following text, the name EMTAC is used to denote either code. A similar modified space-marching approach is used by Rieger.[51]

Single-zone grid generation for a very complex geometry, such as the mated Shuttle orbiter with external tank (ET) and solid rocket booster (SRB), is difficult and requires some configuration approximations. To avoid these geometry modifications, a multizone technique is applied to solve the flow field for this type of geometry.

Fig. 4.4.15 illustrates the multibody interaction problem of the Shuttle orbiter in a mated configuration with the ET and SRBs. The cross-section grids at various constant x-locations are also given. The solutions are obtained at M_∞ = 1.8 and α = 0.0 deg. Fig. 4.4.15 shows the pressure contours from x = 200 in. to x = 1780 in.; five zones are used in this region. The SRB and Shuttle Orbiter are treated as a point, and the zero flux boundary condition is applied to these points. At x = 220, the conical solution of the ET is obtained, and the bow shock is formed.

The solutions are very smooth and continuous across the zonal boundaries despite a very unconventional three-sided computational grid. The expansion wave from the surface at x = 400 in. can be clearly seen in this figure. The apex of the SRB starts at x = 425 in. and produces an attached bow shock. This shock expands and finally hits the ET surface, as can be seen at x = 610 in. Notice that a very symmetric solution is generated by using this unsymmetric grid. The orbiter nose is at x = 640 in. The detached shock and subsonic flow field behind the shock are calculated by using the relaxation method. The reflected orbiter bow shock on the external surface is clearly evident at x = 810 in. At x = 1670 in., the embedded wing shock is indicated which wraps around the leading edge of the wing. Another detached shock is formed further downstream by the orbital maneuvering system (OMS) pod. Since the subsonic pocket is big and the Mach number is almost zero near the root of the OMS pod, a total of 20 relaxation marching sweeps are required to give a good converged result. The OMS pod and vertical tail shocks are clearly shown in this figure at x = 1780 in. The relaxation method is used to calculate the subsonic flow field and detached bow shock. The chordwise pressure distributions on the upper surface of the wing at several span stations compare well with experimental data, including in the region where OMS pod shock interacts with the wing surface.

4.4.7 References

1. RILL, S. and BECKER, K., "Simulation Of Transonic Inviscid Flow Over A Twin Jet Transport Aircraft," AIAA 91-0025, 1991.

2. HOHEISEL, KIOCK, R., ROSSOW, C.-C. RONZHEIMER, A., BAUMERT, W. and CAPDEVILA, H., "Aspects of Theoretical and Experimental Investigations on Airframe/Engine Integration Problems," ICAS-90-2.7.3, 1990.

3. HOHEISEL, H., "Zelle-Triebwerks-Integration - Aufgabe Zukunftiger Verkehrsflugzeugeint- Wicklungen," DLR-Nachrichten, Heft 61, 1990.

4. ROSSOW, C.-C. and RONZHEIMER, A., "Investigations of Interference Phenomena of Modern Wing-Mounted High-Bypass-Ratio Engines by the Solution of the Euler-Equations," AGARD Symposium on Aerodynamic Engine/Airframe Integration, Fort Worth, Texas, 1991.

5. ROSSOW, C.-C., GODDARD, J.-L., HOHEISEL, H. and SCHMITT, V., "Investigations of Propulsion Integration Interference Effects on a Transport Aircraft Configuration," AIAA 92-3097, July 1992.

6. RONZHEIMER, J. A., ROSSOW, C.-C. and PFLUG, M., "Untersuchungen zum Interferenzeinflu__von modernem

Hochbypass-Antriebskonzepten an einer Flugel-Rumpf Kombination durch Losung der Euler-Gleichungen," Proceedings of the DGLR-Jahrestagung Oct, 1990, Friedrichshafen, Germany (in German).

7. TAKANASHI, S., and SAWADA, K., "Numerical Simulation of Compressible Flow Field about Complete ASKA Aircraft Configuration," SAE Publication of the Proceedings of the International Powered Lift Conference P-203, Feb. 1988, pp. 459–466.

8. YU, N. J., KUSUNOSE, K., CHEN, H. C. and SOMMERFIELD, D. M., "Flow Simulations for a Complex Aircraft Configuration Using Euler Equations," AIAA 87–0454, 1987.

9. YU, N. J., CHEN, H. C., SU, T. Y. and KAO, T. J., "Development of a General Multiblock Flow Solver for Complex Configurations," Eighth GAMM Conference on Numerical Methods in Fluid Mechanics, Sept. 1989.

10. KAO, T. J., SU, T. Y. and YU, N. J., "Navier-Stokes Calculations for Transport Wing-Body Configurations with Nacelles and Struts," AIAA 93–2945, 1993.

11. VASSBERG, J. C. and DAILEY, K. C., "AIRPLANE: Experiences, Benchmarks, and Improvements," AIAA 90–2998–CP, 1990.

12. JAMESON, T. J. and BAKER, T. J., "Improvements to the Aircraft Euler Method," AIAA 87–0452, 1987.

13. PEIRO, J., PERAIRE, J., MORGAN, K., HASSAN, O. and BIRCH, N., "The Numerical Simulation of Flow about Installed Aero-Engine Nacelles Using a Finite Element Solver on Unstructured Meshes," Aeronautical Journal, 1992, pp. 224–232.

14. PERAIRE, J., PEIRO, J. and MORGAN, K., "Multigrid Solutions of the 3D Compressible Euler Equations on Unstructured Tetrahedral Grids," Journal of Numerical Methods in Engineering, Vol. 36, 1993, pp. 1029–1044.

15. LECLERCQ, M. P., MANTEL, B., PERIAUX, J., PERRIER, P. and STOUFFLET, B., "On Recent Three-Dimensional Computations Around Complete Aircraft using Adaptive Unstructured Mesh Refinements," Proceedings of the Second World Congress on Computational Mechanics, Stuttgart, Germany, Aug. 1990.

16. SICLARI, M. and DARDEN, C., "CFD Predictions of the Near-Field Sonic Boom Environment for Two Low Boom HSCT Configurations," AIAA 91-1631, June 1991.

17. WHITHAM, G. B., "The Flow Pattern of a Supersonic Projectile," Commun. Pure & Applied Mathematics.

18. WALKDEN, F., "The Shock Pattern of a Wing-Body Combination, Far from the Flight Path," Aeronautical Q., Vol. IX, Pt. 2, May 1958, pp 164–194.

19. MACK, R. J. and NEEDLEMAN, K. E., "The Design of Two Sonic Boom Wind Tunnel Models from Conceptual Aircraft which Cruise at Mach Numbers of 2.0 and 3.0," AIAA 90–4026, 1990.

20. EBERLE, A. and MISEGADES, K, "Euler Solution for a Complete Fighter Aircraft at Sub- and Supersonic Speed," AGARD-CP 412, Paper 17, 1986.

21. EBERLE, A., "3D Euler Calculations Using Characteristic Flux Extrapolation," AIAA 85–0019, 1985.

22. MORGAN, K., PERAIRE, J. and HASSAN, O., "The Computation of 3D Flows Using Unstructured Grids," Computational Methods in Applied Engineering, Vol 87, 1991, pp. 335-352.

23. HETTENA, E., FORMAGGIA, L., ANTONELLINI, A. and SELMIN, V., "Validation of an Unstructured Euler Code for the Simulation of Transonic Flows Past a Complete Aircraft," Proceedings of the Thirteenth International Conference on Numerical Methods in Fluid Dynamics, *Lecture Notes in Physics*, Vol. 414, Springer-Verlag, 1992, pp. 255–259.

24. PERAIRE, J., MORGAN, K. and PEIRO, J., "Unstructured Finite Element Mesh Generation and Adaptive Procedures in CFD," Application of Mesh Generation to Complex 3-D Configurations, AGARD CP-464, 1990.

25. RAJ, P., "Aerodynamic Analysis Using Euler Equations: Capabilities and Limitations," *Applied Aerodynamics*, Progress in Aeronautics and Aeronautics, Vol. 125, pp. 679–700.

26. RAJ, P. and SINGER, S. W., "Computational Aerodynamics in Aircraft Design: Challenges and Opportunities for Euler/Navier-Stokes Methods," International Pacific Air and Space Technology Conference, Gifu, Japan, Oct. 1991.

27. BANGERT, L. H., JOHNSTON, C. E. and SCHOOP, M. J., "CFD Applications in F-22 Design," AIAA 93–3055, 1993.

28. WHITFIELD, D. L., SWAFFORD, T. W., JANUS, J. M., MULAC, R. A. and BELK, D. M., "Three-Dimensional Unsteady Euler Solutions for Propfans and Counter-Rotating Propfans in Transonic Flow," AIAA 87–1197, June 1987.

29. JANUS, J. M. and WHITFIELD, D. L., "A Simple Time-Accurate Turbomachinery Algorithm with Numerical Solutions of an Uneven Blade Count Configuration," AIAA 89–0206, January 1989.

30. JANUS, J. M., "Advanced 3-D CFD Algorithm for Turbomachinery," Ph.D. Dissertation, Mississippi State University, May 1989.

31. JANUS, J. M. and WHITFIELD, D. L., "Counter-Rotating Propfan Simulations Which Feature a Relative-Motion Multiblock Grid Decomposition Enabling Arbitrary Time Steps," AIAA 90–0687, Jan. 1990.

32. HORSTMAN, H. Z., "Incorporation of Radial Partitioning In a Turbomachinery Flow Solver for Unsteady Ducted Applications," M.S. Thesis, Mississippi State University, Aug. 1991.

33. JANUS, J. M., HORSTMAN, H. Z. and WHITFIELD, D. L., "Unsteady Flowfield Simulation of Ducted Prop-Fan Configurations," AIAA 92–0521, Jan. 1992.

34. JANUS, J. M., WHITFIELD, D. L., HORSTMAN, H. Z. and MANSFIELD, F., "Computation of the Unsteady Flow-

field About a Counterrotating Prop-Fan Cruise Missile," AIAA 90–3093, Aug. 1990.

35. JANUS, J. M., MANSFIELD, F. and WHITFIELD, D. L., "A Numerical-Experimental Comparison of Unsteady Counterrotating Prop-Fan Cruise Missile Flowfields," 4th International Symposium on Computational Fluid Dynamics, Davis, CA, Sept. 1991.

36. RANOUX, G., LORDON, J. and DIET, J., "Géométrie at Maillage de Configurations Complexes pour des Calculs Aérodynamiques," AGARD Conference on "Application of Mesh Generation to Complex 3D Configurations," AGARD CP-464, Paper No. 5, May 1989.

37. PARIKH, P., PIRZADEH, S. and FRINK, N. T., "Unstructured Grid Solutions to a Wing/Pylon/Store Configuration Using VGRID3D/USM3D," AIAA 92-4572, 1992.

38. HEIM, E. R., "CFD Wing/Pylon/Finned Store Mutual Interference Wind Tunnel Experiment," AEDC-TSR-91-P4, Jan. 1991.

39. DESLANDES R. M., "An Overlapping Technique for Steady and Unsteady Interference Problems," MBB Publication S–484, 1992.

40. HEIB, S., "Oberflachennetzgenerierung aus Einem CAD-Modell mit Einer Neuen Projektionstechnik," MBB-FE211 Publication AERO/MT/913, 1992.

41. DESLANDES R.M., "Theoretische Bestimmung der Trajektorien abgehender Flugkorper von Kampfflugzeugen," MBB UFE 1361, 1978.

42. DOUGHERT, F.C., et al., "Transonic Store Separation Using a 3D Chimera Grid Scheme," AIAA 89-0637, 1989.

43. DESLANDES R. and EBERLE A., "Characteristic Flux Averaging Approach to the Euler Equations (Part II)," MBB-LKE/12/S Publication 313, 1987.

44. RIEGER, H., STOCK, H. and WAGNER, B., "Computational Aerothermodynamic Methods For Industrial Applications To Reentry And Hypersonic Cruise Problems," ESA-SP-318 (1991), pp. 421-437.

45. SCHONE, J. and BIDAULT, J., "Calculations of Inviscid Flow Around A Reentry Configuration at Supersonic Speed," AIAA-91–0391, 1991.

46. SCHONE, J., "Aerodynamische Berechnungen fur eine HERMES- Studienkonfiguration bei Uberschallanstromung durch Losung der Euler-Gleichungen," Proc. of the DGLR-Jahrestagung, Oct. 1-4, 1990, Friedrichshafen, Germany.

47. PFITZNER, M., "Runge-Kutta Split-Matrix Method for the Simulation of Real Gas Hypersonic Flow," Proceedings of the Second International Conference on Nonlinear Hyperbolic Problems, *Notes in Numerical Fluid Mechanics*, Vol. 24, 1989, pp. 489–498.

48. MENNE, S. and WEILAND, C., "Calculation of Three-Dimensional Viscous and Inviscid Hypersonic Flows Using Split-Matrix Marching Methods," AIAA 90–3070, 1990.

49. CHAKRAVARTHY, S. R. and SZEMA, K. Y., "An Euler Solver for Three-Dimensional Supersonic Flows with Subsonic Pockets," AIAA 85-1703, 1985.

50. SZEMA, K. Y., CHAKRAVARTHY, S. R., RIBA, W. T., BYERLY, J. and DRESSER, H. S., "Multi-Zone Euler Marching Technique for Flow over Single and Multi-Body Configurations," AIAA 87–0592, 1987.

51. RIEGER, H., "Solution of Some 3D Viscous and Inviscid Supersonic Flows by Finite-Volume Space Marching Scheme," AGARD CP-428, Paper No. 17, 1987.

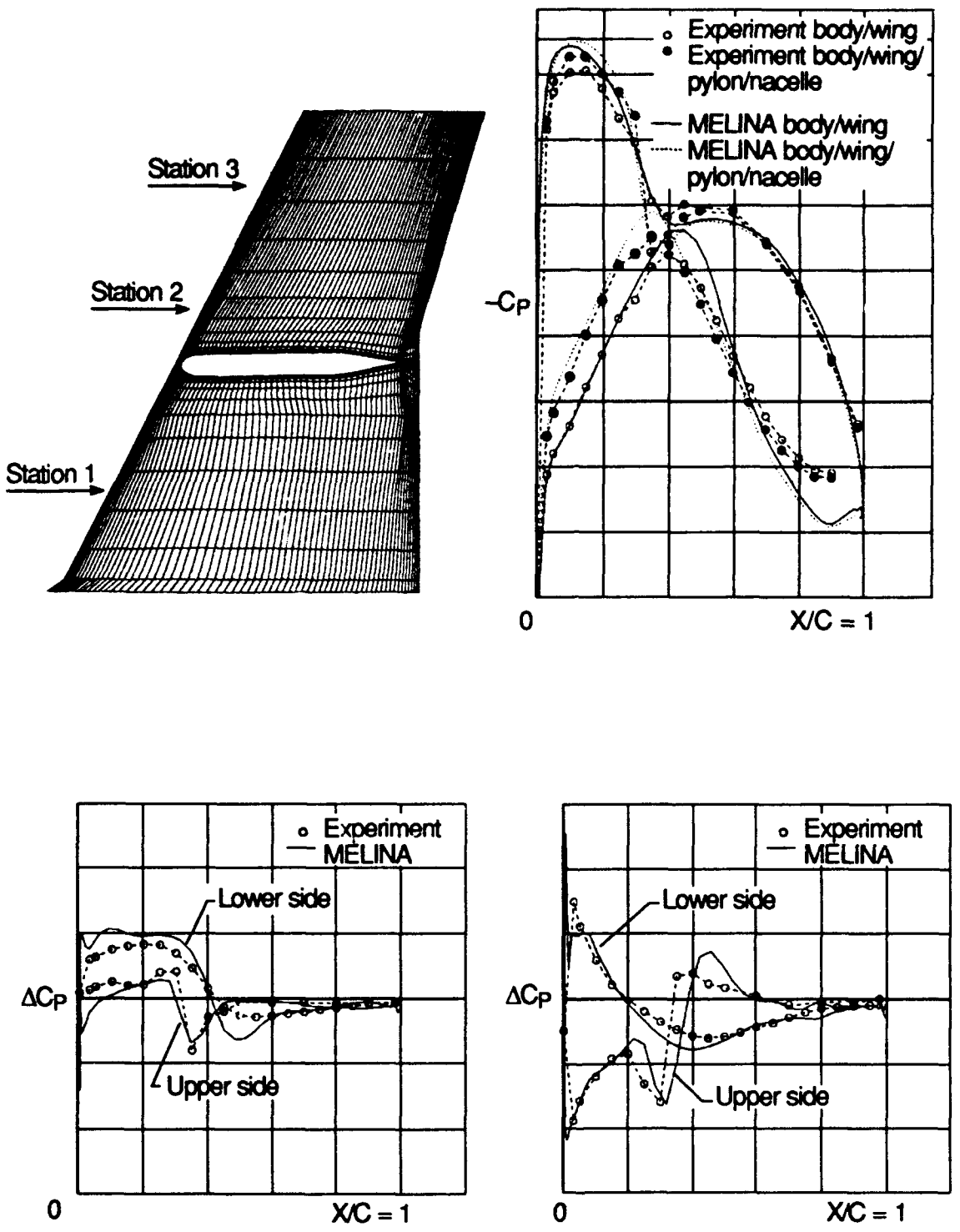
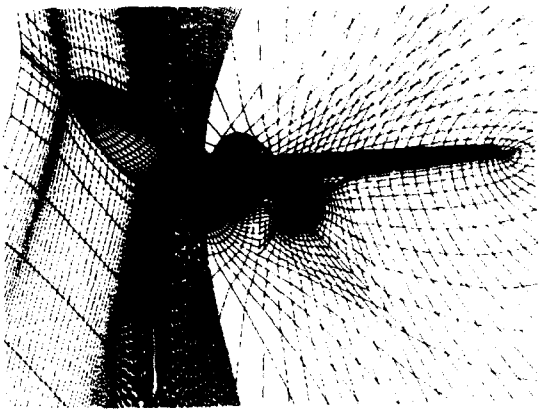
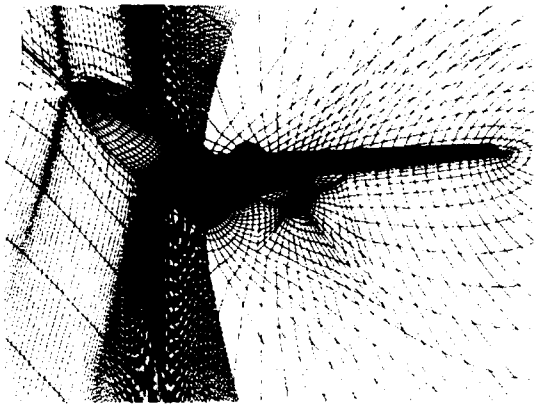


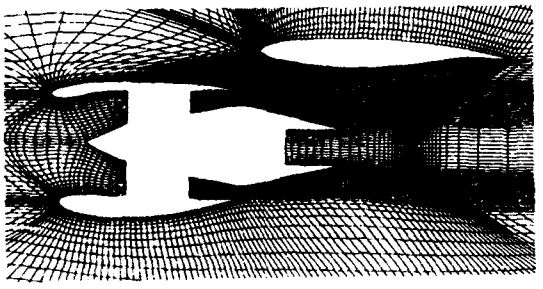
Figure 4.4.1 Euler simulation of transonic flow over a twin-jet transport aircraft.



Wing-body + UHB



Wing-body + CFM 56



Cut through wing and engine

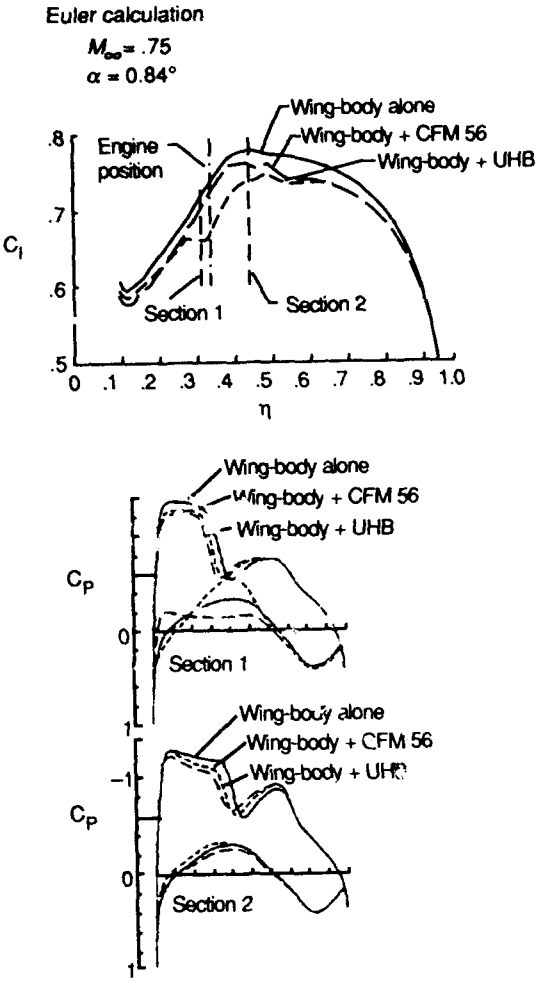
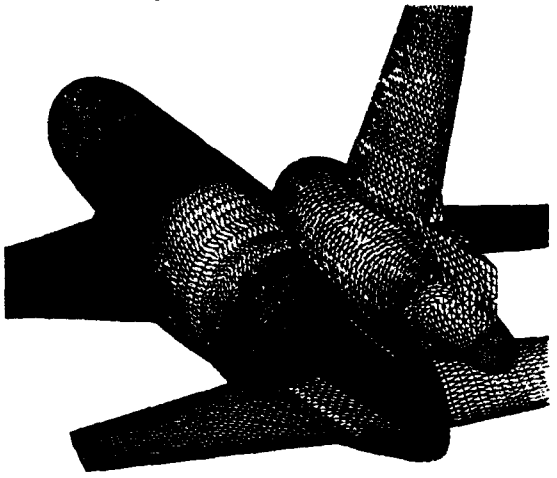
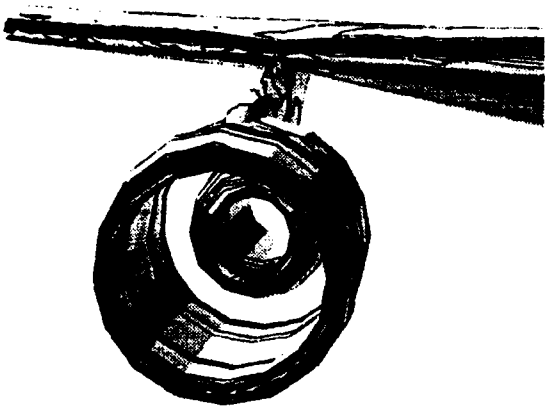


Figure 4.4.2 Engine-airframe integration for a transport aircraft with an ultra-high bypass engine.

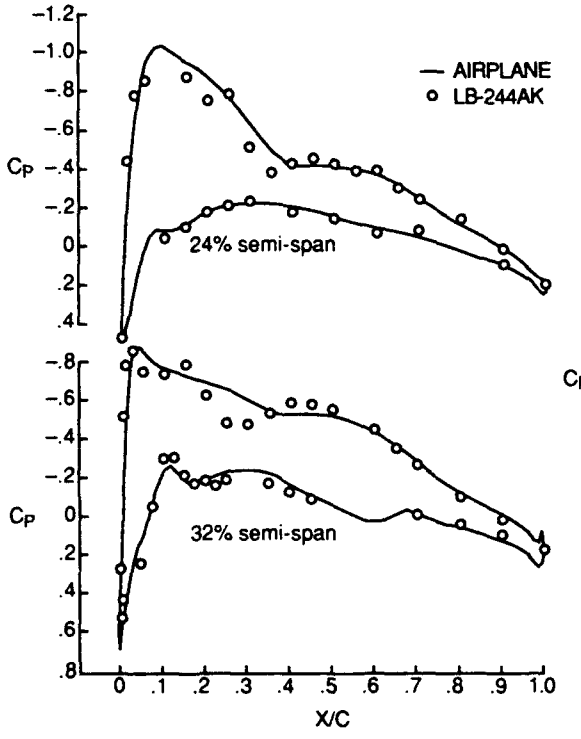
Surface triangulation of the MDC tri-jet configuration



Pressure contours of the under-wing engine



Comparison of computed pressures with test data



Comparison of pressures with and without engine assembly

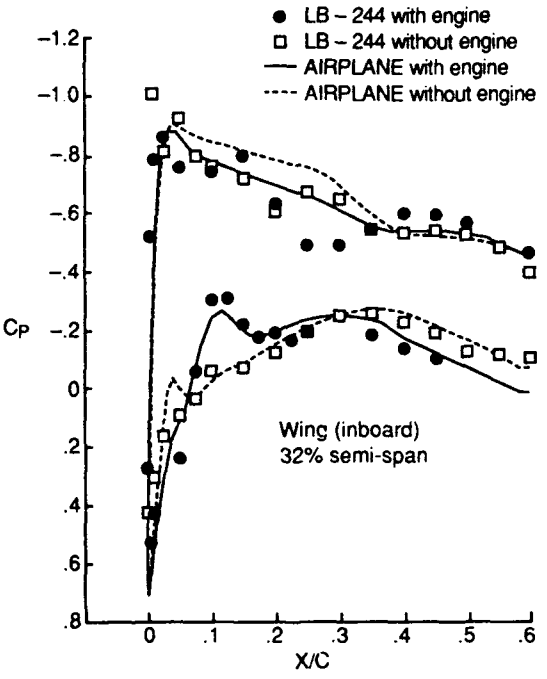


Figure 4.4.3 Unstructured–grid applications to a McDonnell–Douglas Tri-Jet configuration at transonic speeds; M_∞ = 0.825, α = 1.0 deg.

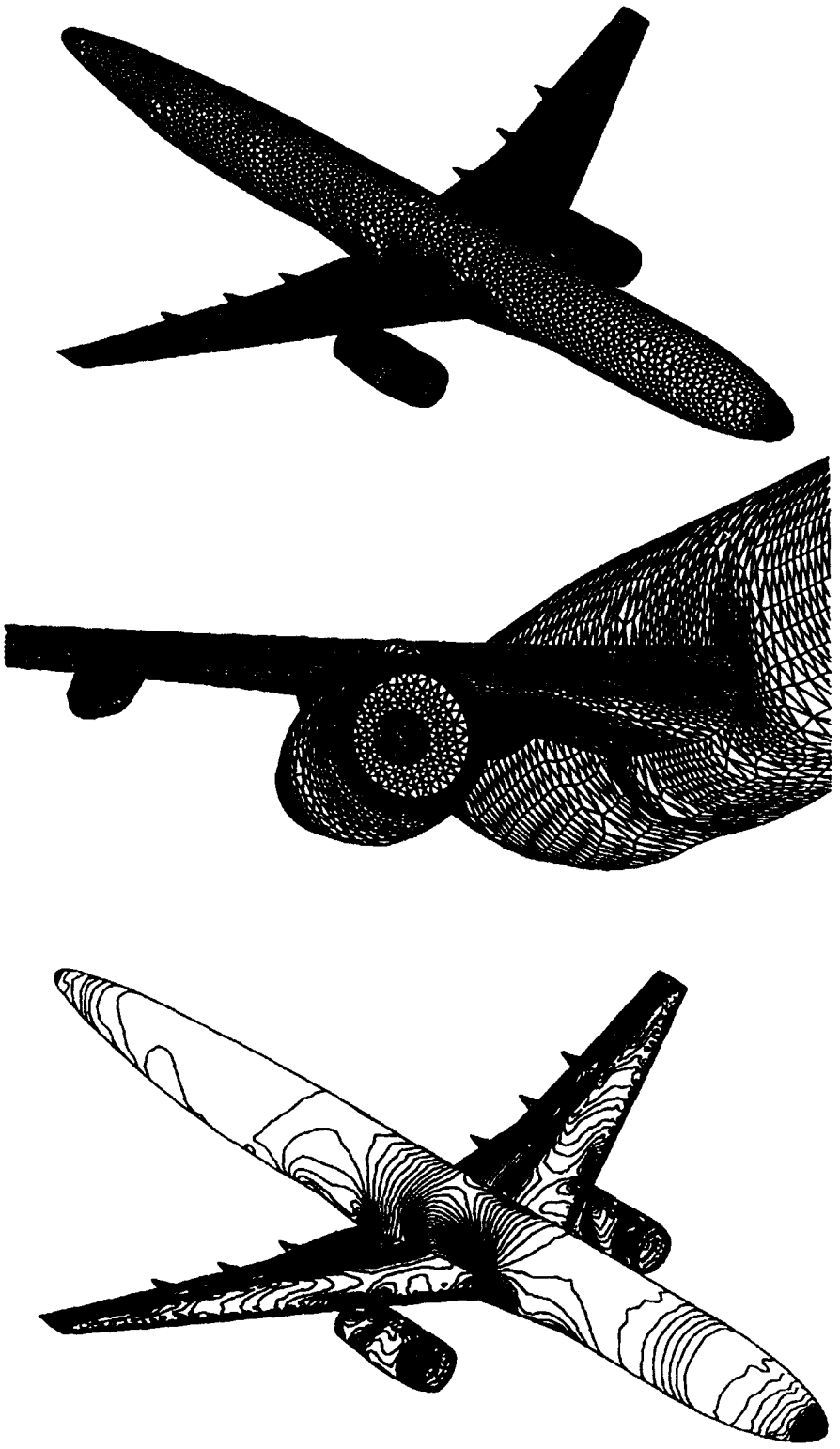
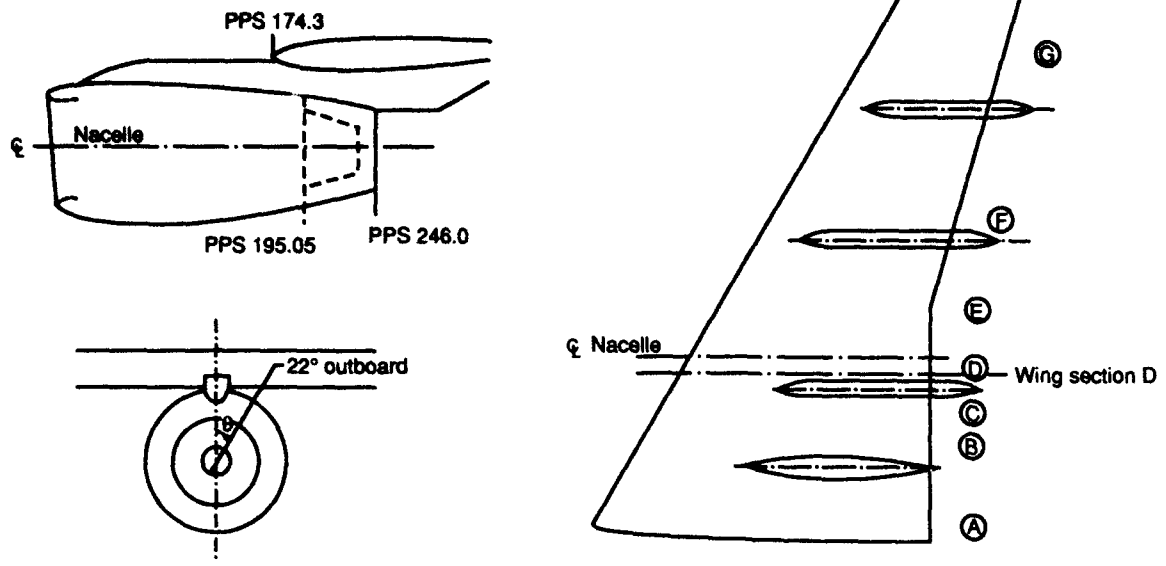
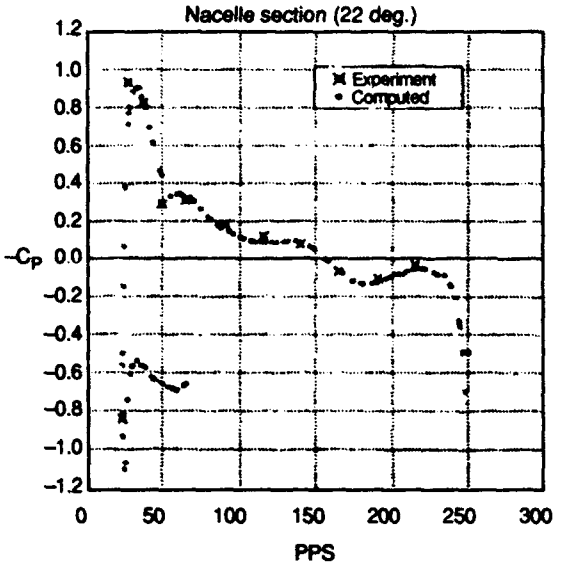


Figure 4.4.4 Unstructured surface grid and pressure contours for a transport configuration; M_∞ = 0.801, α = 2.738 deg.





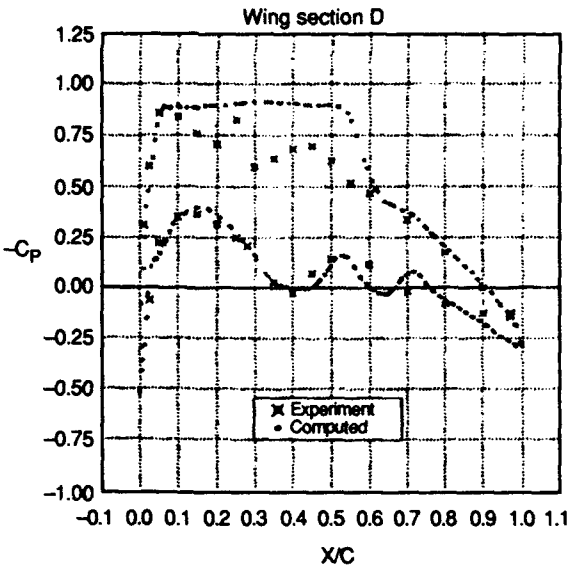


Figure 4.4.5 Comparison of streamwise pressure distributions along nacelle and wing surface between Euler and experiment; M_∞ = 0.801, α = 2.738 deg.

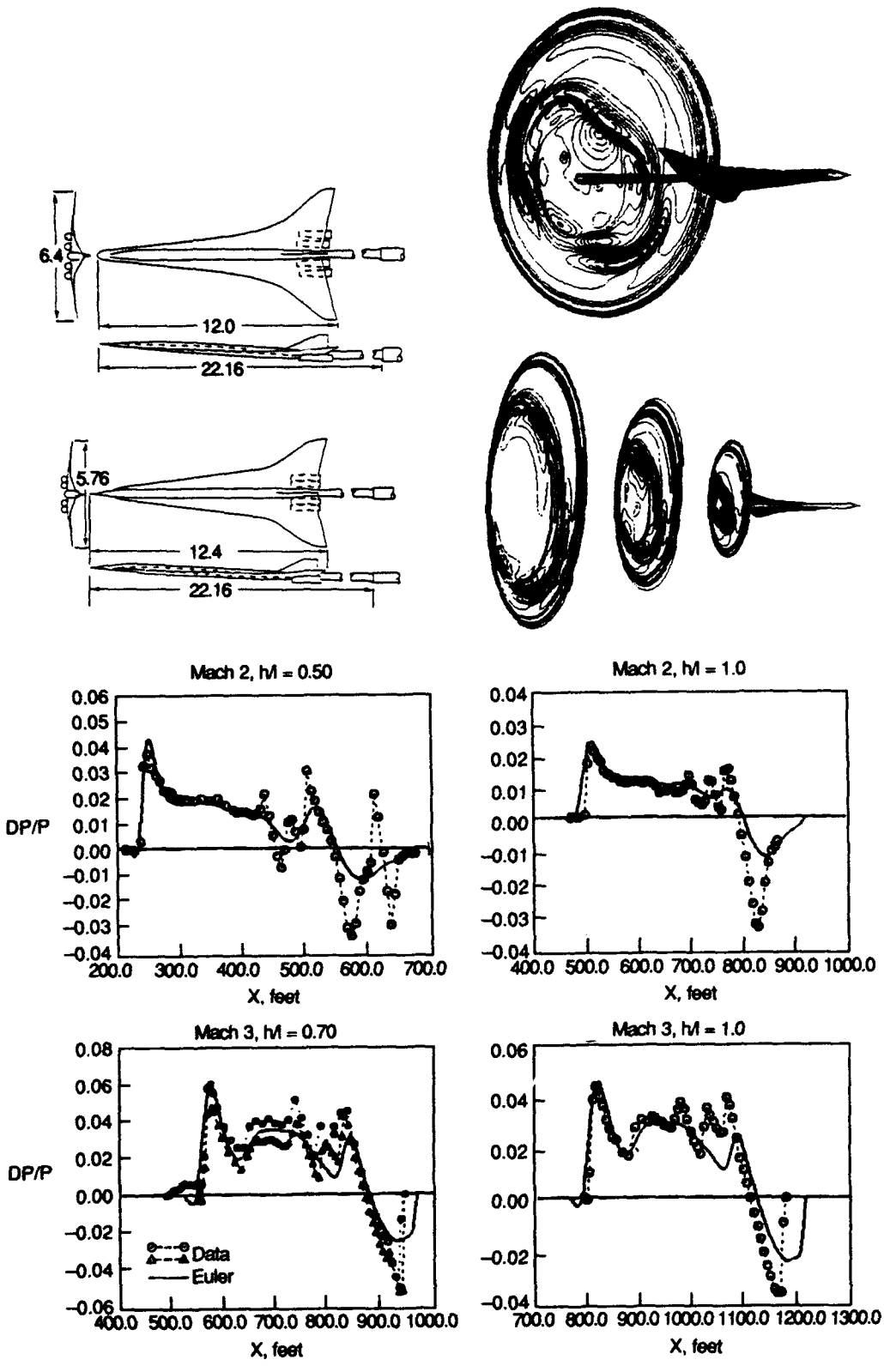
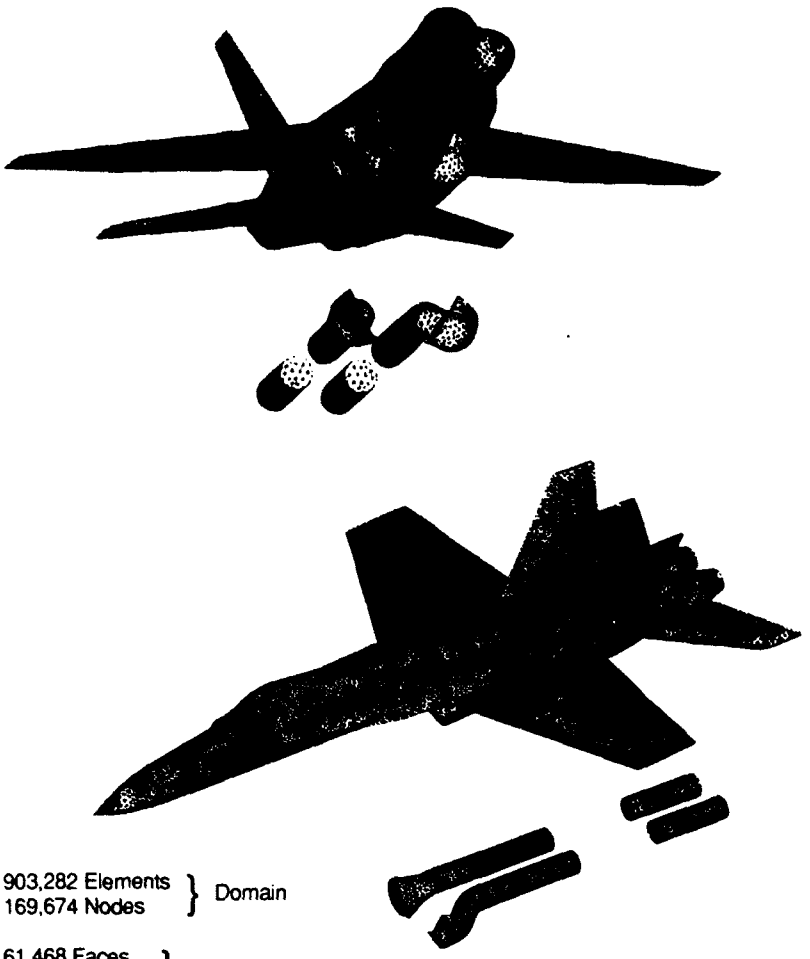


Figure 4.4.6 Euler computations and comparisons with experiment for two supersonic transport configurations.



903,282 Elements
169,674 Nodes } Domain

61,468 Faces
30,738 Nodes } Surface

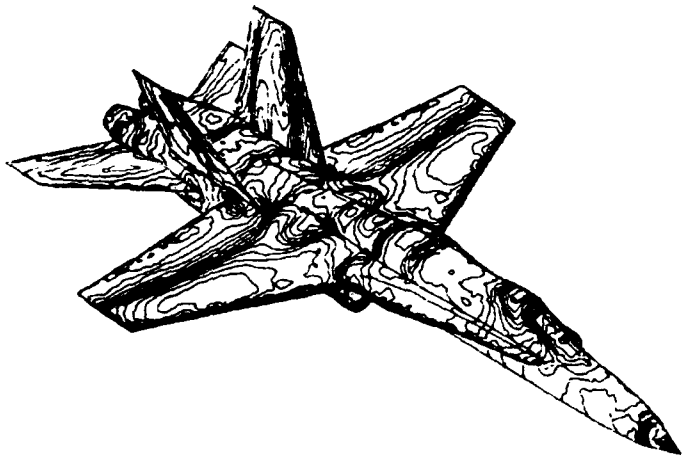


Figure 4.4.7 Unstructured surface grid and pressure contours for an F-18 aircraft configuration; M_∞ = 0.9, α = 3 deg.

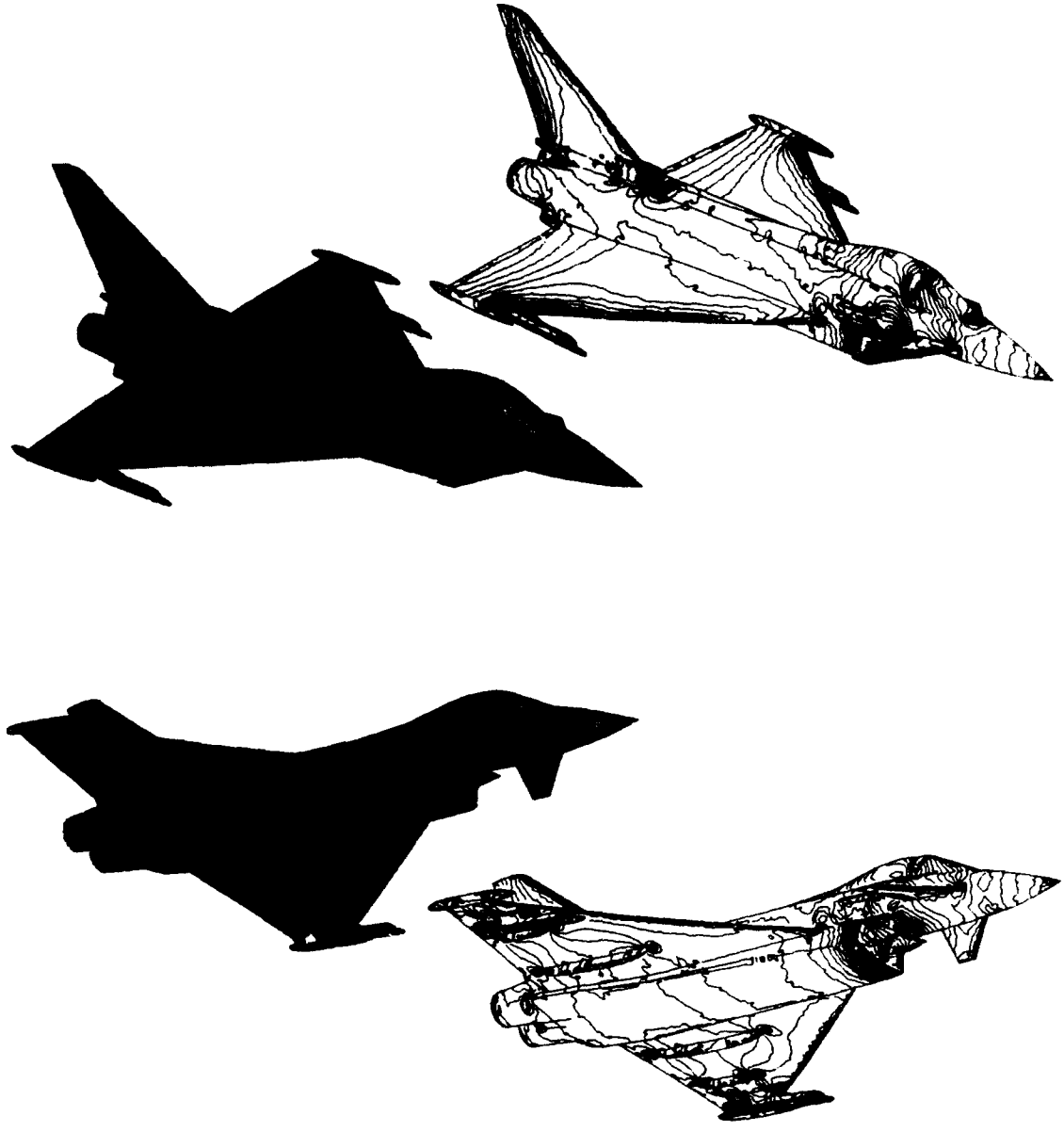
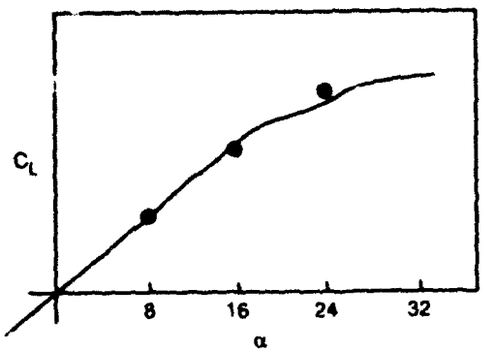
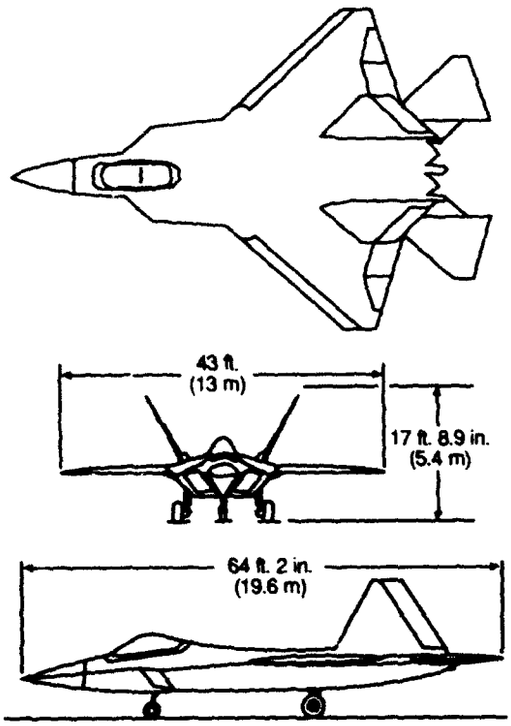


Figure 4.4.8 Surface grids and pressure contours from an unstructured-grid computation of an advanced fighter configuration.



Presure comparisons at $\alpha = 8°$ for two stations on the wing--one near midspan and the other near the tip.



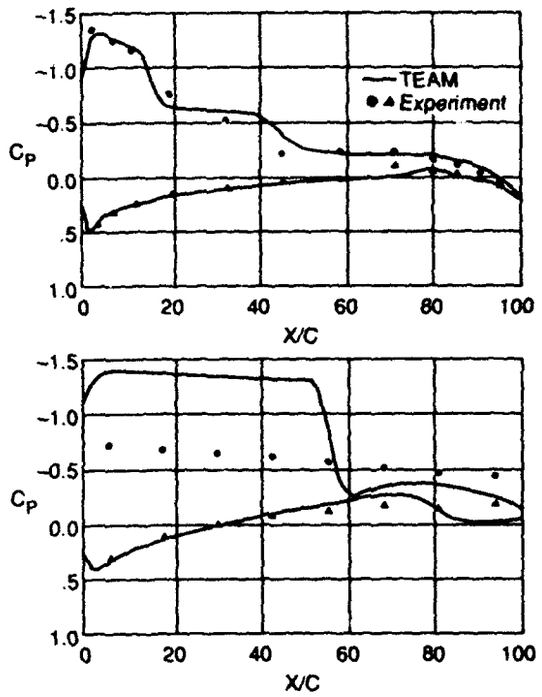
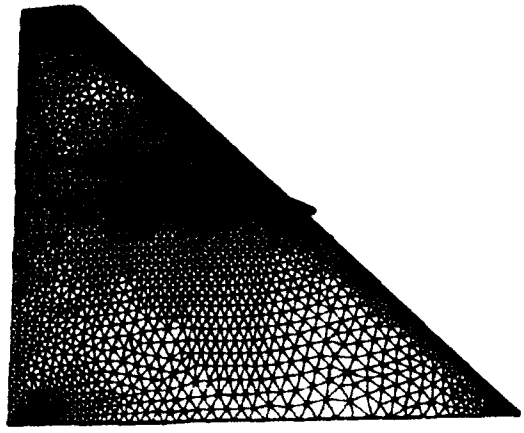
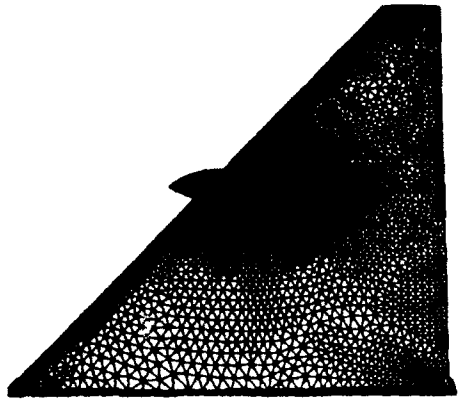


Figure 4.4.9 *Multiblock structured-grid Euler computations for advanced tactical fighter configuration.*

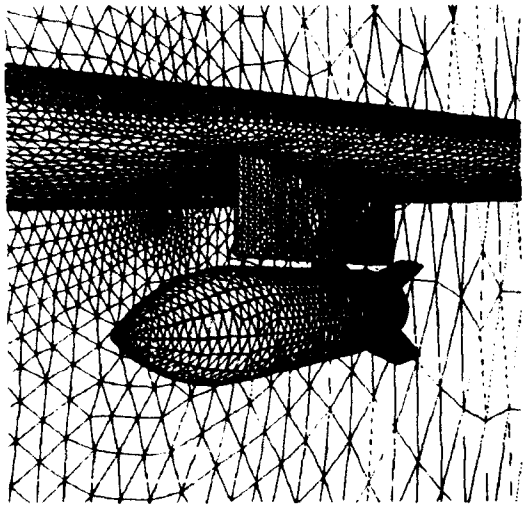


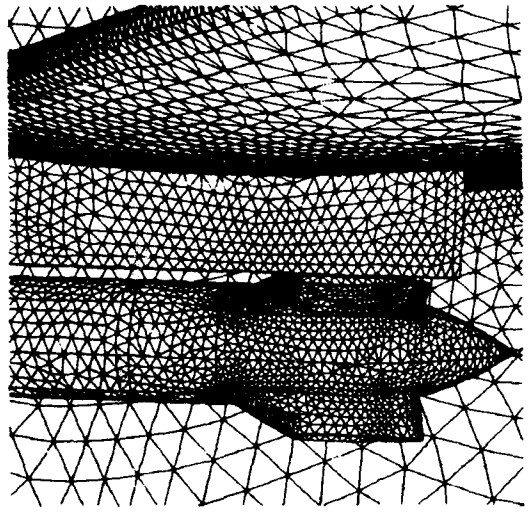
Upper surface



Lower surface

On the wing





Close-ups

Figure 4.4.10 Surface triangulation on a wing/pylon store configuration.

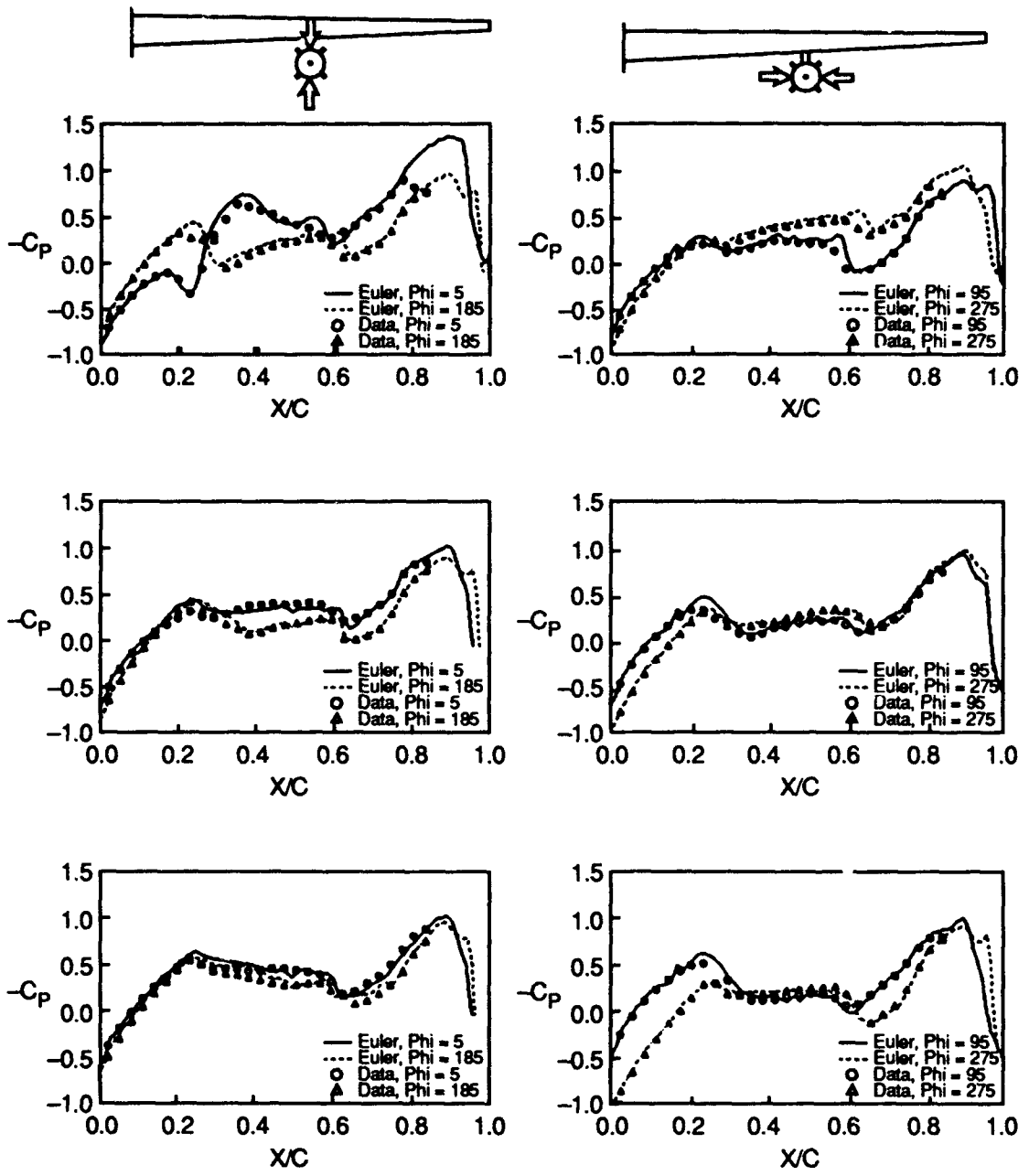


Figure 4.4.11 Surface pressure comparison on the store of a wing/pylon store configuration, M_∞ = 0.95.

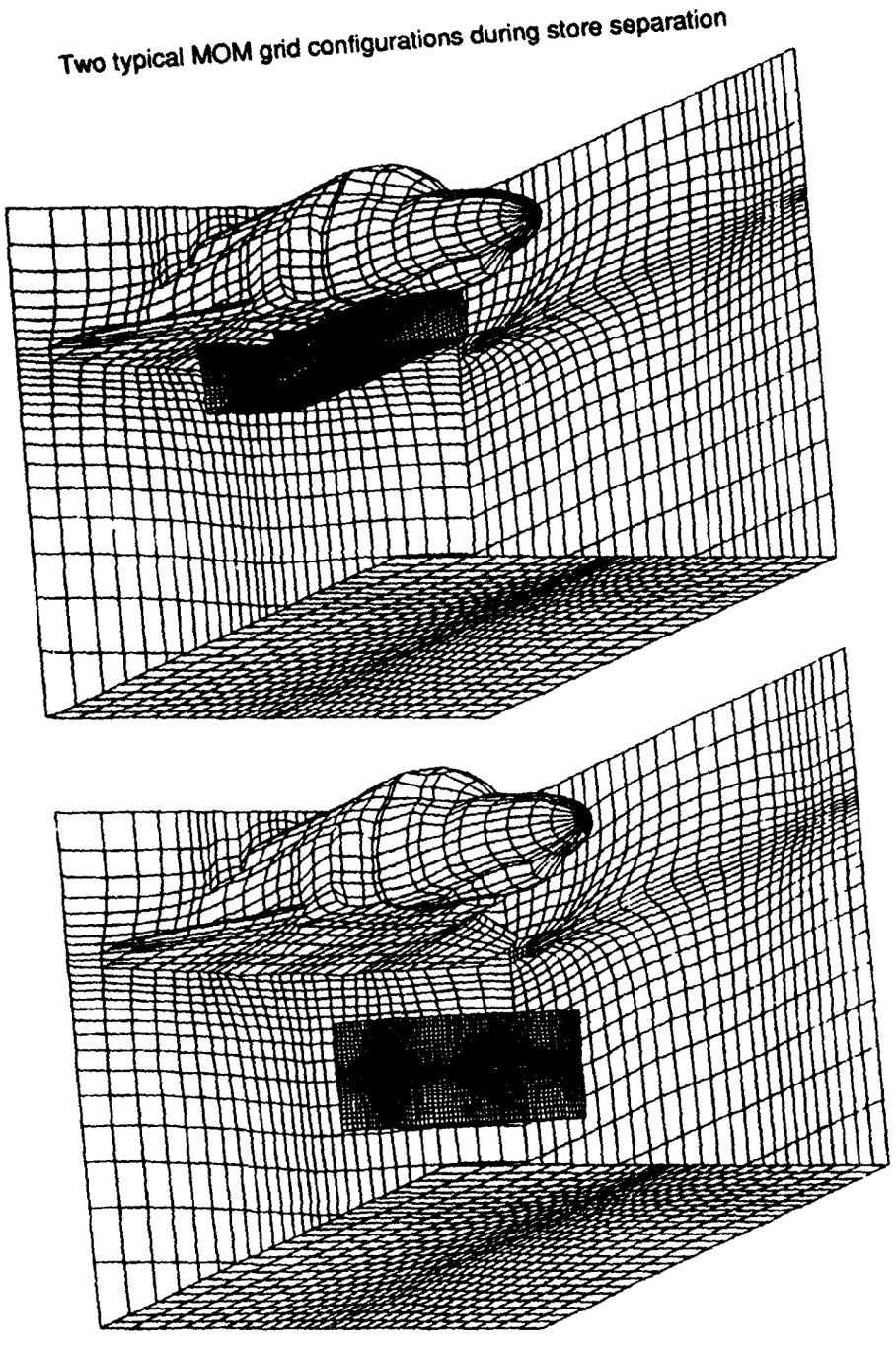


Figure 4.4.12 Two typical multiblock structured grids developed during the simulation of a store separation from a forebody configuration.

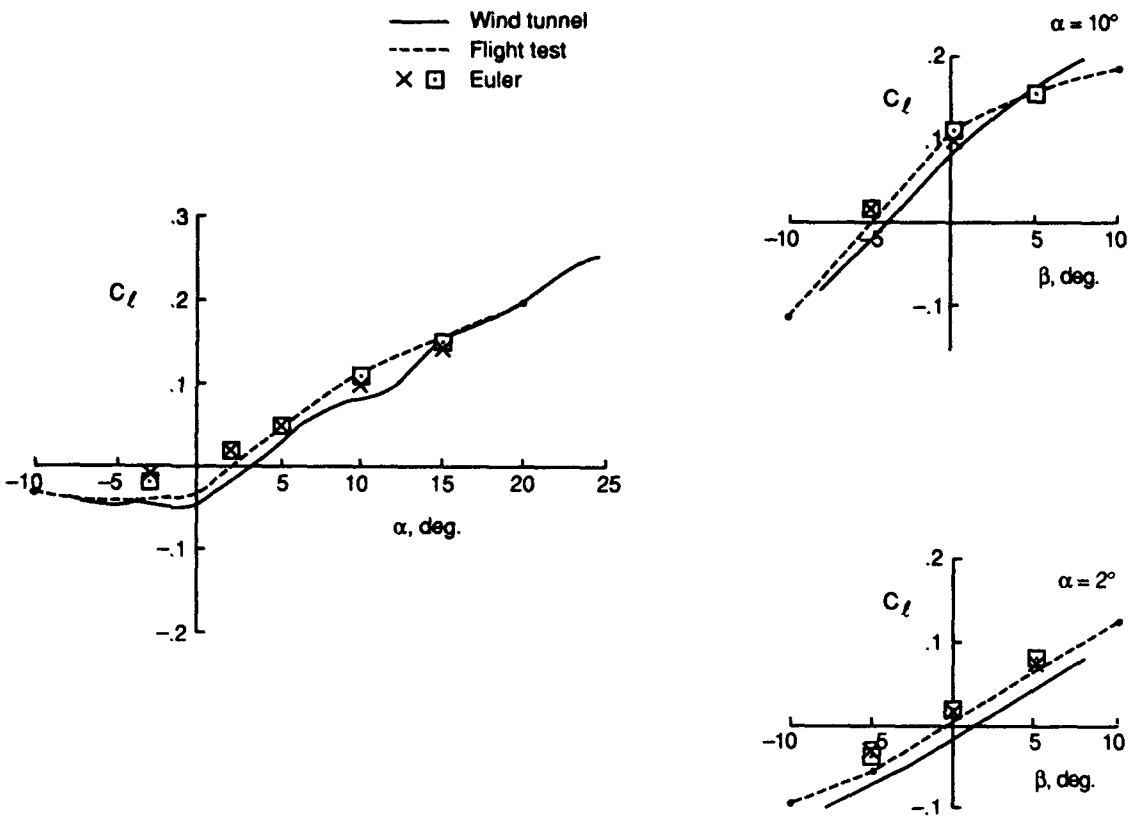


Figure 4.4.13 Comparison between Euler (MOM), flight-test (FT), and wind-tunnel test (WT) for angles of attack and sideslip; M_∞ = 0.9.

Selected grid planes for half of the symmetric configuration

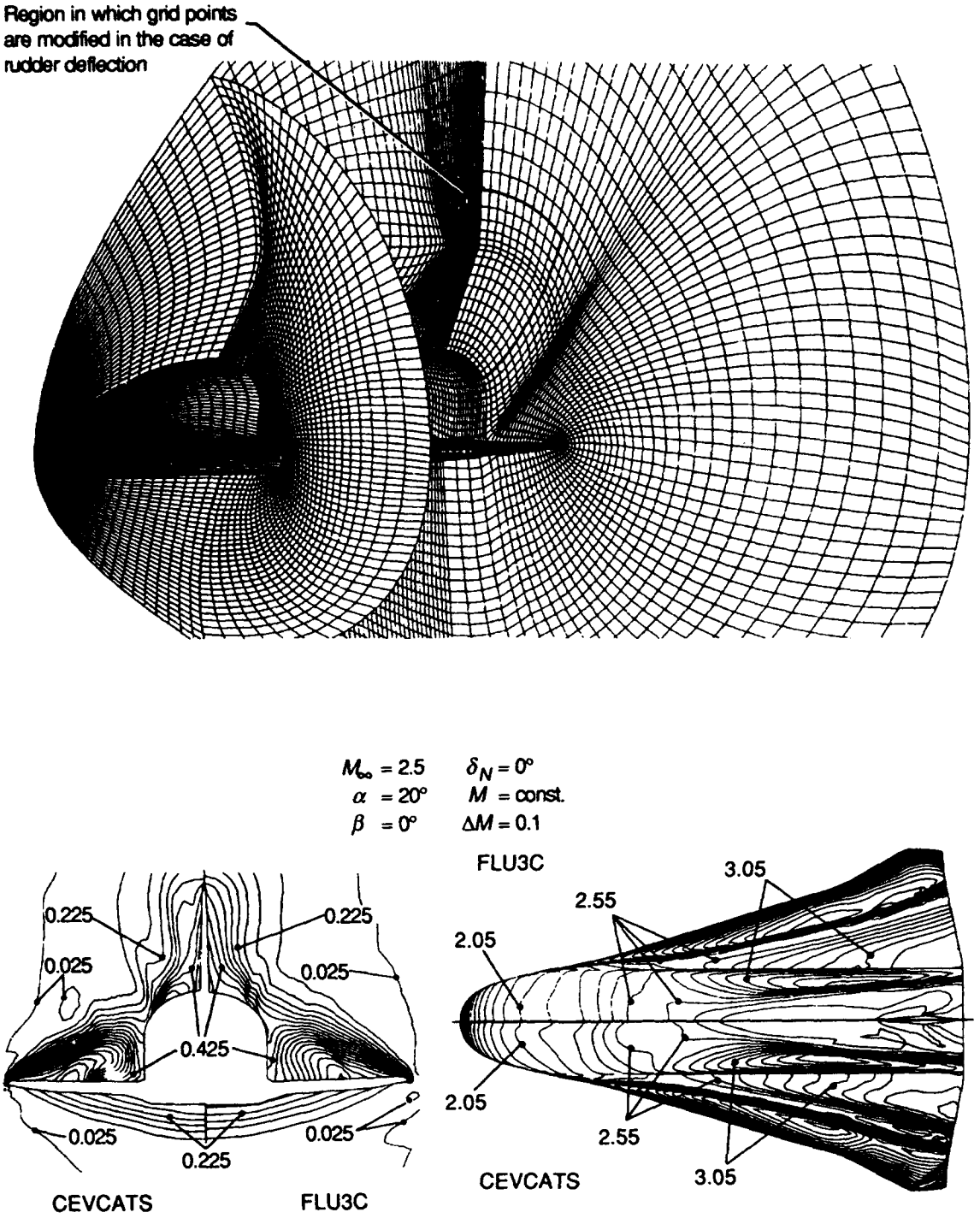


Figure 4.4.14 Euler computations for a hypersonic reentry configuration.

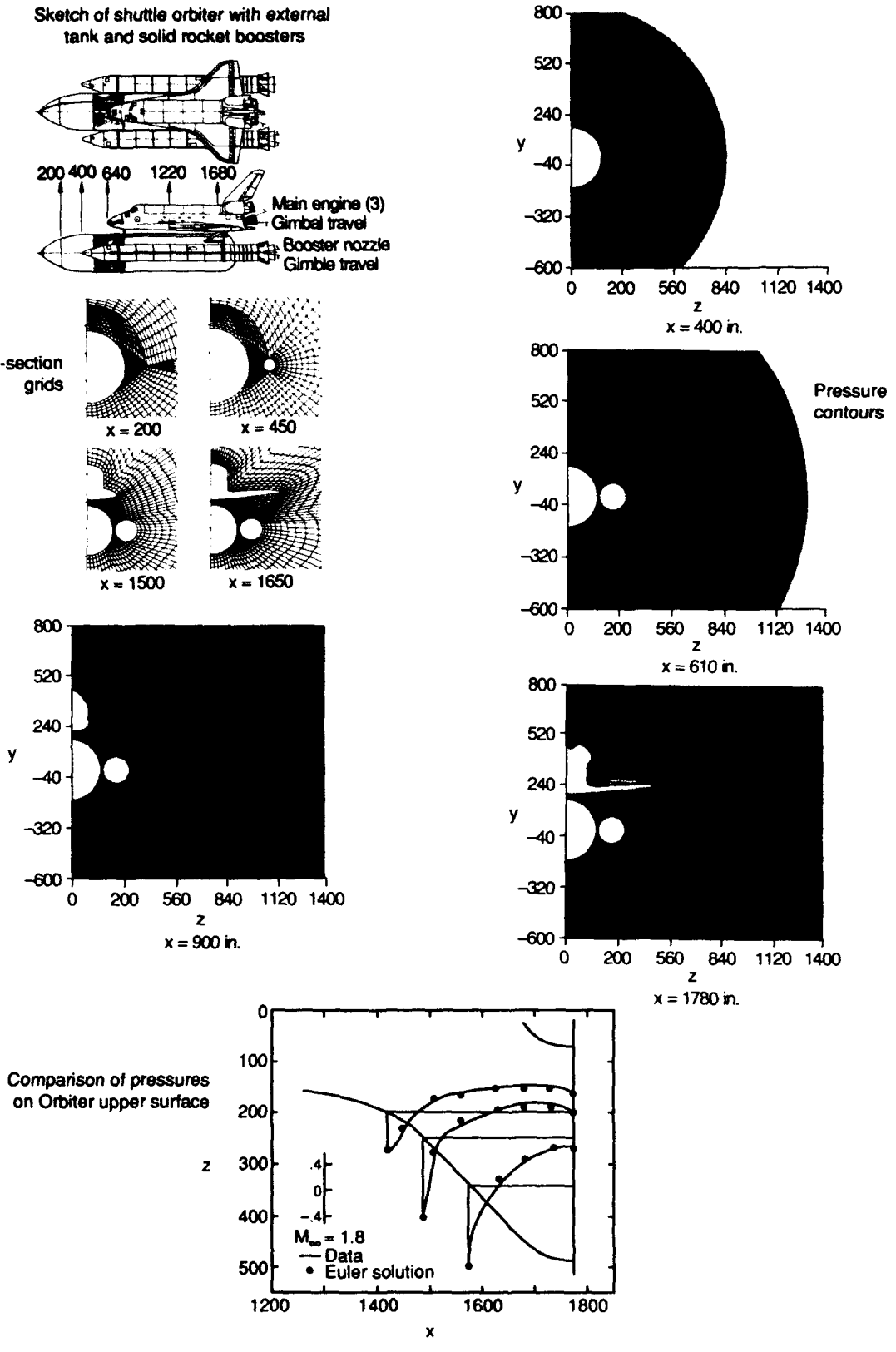


Figure 4.4.15 Multi-zone Euler marching method applied to the Space Shuttle orbiter-tank-booster configuration; $M_\infty = 1.8$, $\alpha = 0.0$ deg.

4.5 PROPULSION SIMULATIONS

Applications of the Euler equations to simulate propulsion flow fields are shown below. The Euler equations have been used extensively in propulsion simulations and propulsion-airframe integration studies. These equations admit vorticity and, thus, can be used to simulate the aerodynamic performance (neglecting boundary-layer effects) of rotating machinery such as propellers, rotors, or turbines. Simulations of propeller, rotor, cascade, and fan blade passage calculations are shown below, including simulations in which an approximate model is used to determine the inflow conditions to a local Euler solution and several more detailed computations in which the entire flow is modeled. Because of the geometrical complexity of the three-dimensional, generally unsteady flow fields, and the difficulties in resolving free wakes in the flow field, the computational demands associated with the latter classes are high; such calculations are yet in their infancy. The modeling of the propulsion system is often done using an actuator disk model, admitting specified changes in total pressure, enthalpy, and rotation, and an example for a twin tractor configuration is shown; several other airframe-integration examples shown previously also used actuator disk modeling. Euler computations for nonrotating machinery, such as inlets, nozzles, and nacelles are comparatively much more evolved; several detailed computations are shown, including a nacelle analysis system which has been extended into the design stage (Section 4.8) and several higher Mach number applications. At high Mach numbers, simulations of inlets and nozzles associated with high-speed aircraft must consider the Euler equations at a minimum since the potential equations are inadequate.

4.5.1 NACA Propeller

Figure 4.5.1 shows results[1,2] of a calculation for a NACA propeller with $M_\infty = 0.56$, an advance ratio of $\Lambda = 0.73$, and a blade tip Mach number of $M_{tip} = 0.96$. Lines of constant pressure are plotted in four crossplanes perpendicular to the blade in the upper part of the figure. A shock can be seen in the outward planes for $z/Z_s > 0.7$, where Z_s denotes the radial position of the blade tip. This shock is getting stronger and its position is getting closer to the trailing edge with increasing radius z. This effect is clearly represented in the lower part of Fig. 4.5.1, which shows the lines of constant pressure on the blade's upper surface. The comparison between calculated and measured pressure distributions along the chord for the same case at four selected cross sections shows good agreement. The calculation produces a shock that is stronger and its position is more downstream than the experiment, which is attributed to viscous effects.

4.5.2 Hovering Rotor

Approximate Wake Model

Figure 4.5.2 shows a comparison between computation[1,2] and experiment[3] for a two-bladed rotor with untwisted rectangular NACA 0012 blades in hover. The blade tip Mach number M_{tip} is 0.794 and the collective angle of attack θ is 8 deg. The pressure distributions along the chord at four different cross sections are presented. These flow conditions produce a transonic flow in the region of the blade tip. The agreement between computation and experiment is good in all four sections. The position and strength of the shock are well reproduced by the numerical algorithm. Other computations are given by Stahl.[4]

Complete Wake Model

In the model described above, only a small region around the blade has to be discretized since a wake model is used to set the inflow conditions for the blade. Consequently, the flow induced by the advancing blade is not the flow that impacts the following blade. This leads to large savings in computer time but the solution is dependent on the wake model used. To remedy this deficiency, the physical domain can be extended so that the complete rotor disk is enclosed. This ensures that the following blades are exclusively exposed to an induced flow that is calculated by the pure Euler procedure itself. The problem associated with this is a further increase in the number of grid cells and, consequently, in the computational expense. On the other hand, the main function of the cells inside the expanded regions is only to provide the essential distance between the blade and the far field boundary. Since the flow gradients that are expected in these regions are less substantial than in the blade's vicinity and their influence on the blade is small, the discretization can be much coarser than in the interior.

Using this approach, Kramer et al.[5] found that if the grid is too small, especially in the radial and downward directions, the development of the rotor wake is obstructed significantly. To overcome this deficiency, the dimensions of the original grid (twelve blade chords, which is twice the blade length, in the radial direction, and eight blade chords above and below the rotor disk) were enlarged by a factor of 2-3 in the radial and 3-4 in the normal direction. The original grid is actually used as an inner component of the extended grid, generated separately by a simple analytical algorithm. An example for an O-O grid generated in this manner is given in Fig. 4.5.3.

An Euler calculation based on a free stream initial condition corresponds to the physical situation of a helicopter rotor starting instantaneously from rest. The flow behavior associated with the beginning of rotation is very complex and characterized by a long transient period before a steady state is reached. As lift develops, the starting vortex created by the rotor blades plays a dominant role. The formation and the further temporal evolution of the rotor wake within the starting phase is known from several experimental investigations.

Figure 4.5.3 shows the results obtained in terms of the circulation per unit area using an expanded O-O grid. The temporal evolution of the wake within the starting phase is plotted at different iteration levels. The figures clearly show the movement of the tip vortex, as well as that of the vortex arising at the inner blade root. The reproduction of the inner vortex and the highly three-dimensional interference between the inner and the outer vortex is not possible unless the physical domain completely extends to the rotor axis. If the region of small radii was not considered, as is often done in practice, these effects would be ignored.

The results of the Euler procedure show the typical phenomena of the starting process as in the experiments: initially, the vortex ring remains located near the tip, being continuously fed with circulation. Likewise, its geometric extension continues to grow. After some time, the vortex begins to descend and new vortices arise. This is indicated by the isolines

becoming more and more elongated. With the extended grid, however, there is no impeding boundary and the movement proceeds undisturbed. Some time later, one can see that the isolines are directed inwards, which is a typical sign of the beginning contraction of the newly arising vortices. This process intensifies as the starting vortex ring retains its radial position. In the last frame, the starting vortex is approximately 15 blade chords beneath the rotor disk. At this distance, any influence on the blade has been lost.

A quantitative proof of the agreement between calculation and experiment can only be performed for the time-asymptotic state because a steady code was used to calculate the transition phase and, therefore, time consistency does not exist. The geometric locations of the outer tip vortices for the steady case corresponding to a collective pitch of 5 deg and a tip Mach number of 0.815 is plotted into the isoline diagram of the final iteration state. These results are in a very good qualitative agreement with the experiments. The agreement between the numerical and the experimental data is very good, except at the very tip, where the tip release is overestimated. The reason for this is not yet clear, since the discrepancy could not be improved by using the expanded grid and was also unaffected by any grid refinement, geometric tip modeling, or other measures of that kind.

Figure 4.5.3 also shows the results obtained in terms of the pressure distribution for the two-bladed model rotor of Caradonna and Tung[1] for a tip Mach number of $M_{tip} = 0.815$ and a collective pitch angle of $\theta_c = 5$ deg. The measured pressures agree closely with the predictions.

4.5.3 Turbine and Compressor Cascades

The reliability of modern axial-flow turbomachines is particularly influenced by flow-induced vibrations. They may be caused through blade row interactions, turbulence, stall, inlet distortion, and self-exited blade vibrations.

Based on the explicit predictor/corrector MacCormack scheme,[6] a numerical method has been established for two-dimensional computation of unsteady inviscid subsonic flow through oscillating compressor and turbine cascades by the Euler equations.[7,8] As a typical result for an unsteady oscillating blade application of the numerical method, the travelling wave mode (TWM) of a compressor cascade in bending motion is investigated and compared to corresponding measurements. In TWM, all blades vibrate at the same frequency and amplitude, but a circumferential constant phase lag exists between each blade and its neighbors.

A computational mesh (51x17 points) typical for such an application is shown in Fig. 4.5.4. The geometry presented corresponds to the standard configuration no. 4 as defined for the 1986 workshop[9] on "Aeroelasticity in Turbomachines" organized by the EPFL-Lausanne. In the numerical procedure, the H-type mesh is smoothed at each time step by an elliptic operator in order to improve convergence characteristics.

The location of the various numerical boundaries with special algorithmic treatments are depicted in Fig. 4.5.4. As indicated, the following boundary types have to be distinguished: inflow (B1) and outflow (B2), wall (B3, B4), slip (B5, B6), and periodical boundary in circumferential direction (B7, B8). Downstream of the blade, a slip-line leaves the trailing edge which allows the representation of a discontinuity in the tangential velocity field for simulation of vorticity effects.

The numerical results are compared with experimental data in Fig. 4.5.4. The profiles of the compressor cascade were equipped with five pressure transducers on the pressure side (D1–D5) and six pressure transducers on the suction side (S1-S6). For these locations, experimental and numerical data in the form of the dynamic pressure distributions are plotted as a function of time. The comparison with computed results shows very good agreement in amplitude and phase except for D1. Unfortunately, the pressure transducer S1 was out of order. The acceleration measured in the experiment shows a second, higher frequency corresponding to the first torsional mode. Therefore, all measured data exhibit a high oscillation. The difference at point D1 is caused by the relatively coarse grid. The good calibration between measurement and numerical simulation with respect to phase shifts can be examined by the correspondence of acceleration data.

Other Euler computations of the internal flow in cascades, turbines, and ducts are reported by Saxer et al.,[10] Happel et al.,[11,12] Lecheler et al.,[13] and Leicher.[14]

4.5.4 Fan Stage Passage

A cooperative program between Mississippi State University and the National Aeronautics and Space Administration (NASA) has been in place for 6 years to develop software capable of the time-accurate analysis of complex rotating machinery.[15-18] Recently, NASA's interest has shifted toward ducted rather than unducted advanced turboprop designs; consequently the computational fluid dynamic (CFD) effort was steered to extend the flow analysis software developed for the time-accurate simulation of unducted geometries (prop fans) by incorporating additional domain decomposition mechanisms to enable the simulation of unsteady ducted prop-fan flows (i.e., combined external and internal flow).[19] This effort uses computational techniques and experience gained in computing unsteady flows about complex geometries using dynamic multiblock grids (i.e., relative-motion subdomains). Although initially intended for the numerical solution of rotating machinery problems, the computational tools that were developed at Mississippi State essentially comprise a structured multiblock flow solver and have been used for the flowfield simulation of a complete aircraft configuration, such as the prop-fan powered cruise missile shown in a previous section. The references cited present a detailed discussion of the numerics of the flow solver which includes the equation formulation (finite volume), the numerical flux at cell faces for this cell centered scheme (flux-difference split with Roe averaging), and the implicit solution algorithm (block LU approximate factorization with iterative refinement), along with a discussion of the dynamic multiblock grid approach, including techniques that have been developed for this particular type of problem involving rotating blocks. One configuration presently undergoing investigation is a 1.15 pressure ratio fan stage extensively tested by NASA.[20] The 225x52x15 H grid was used to model one-twelth of the geometry, one rotor passage, and three stator passages (benefitting from solution symmetry). The configuration operates at an advance ratio of 2.86 with a free stream Mach number of 0.75. The predicted surface pressure contours of the present Euler flow solver are

shown in Plate 4.1. Other three-dimensional Euler computations for a propeller are given by Bocci et al.[21]

4.5.5 Propeller Slipstream

An example[22] indicating the use of Euler equations to model propeller-slipstream effects is shown in Fig. 4.5.5. The surface grid for a high-wing transport aircraft configuration intended for subsonic flight is shown; two wing-mounted propellers mounted in the tractor position provide thrust. The propeller is modeled as an actuator disk in the computations and generates a slipstream having swirl and increased total pressure. The calculations were done using a multiblock grid consisting of 1.2 million cells and solved with an explicit central-difference discretization. A top view of the surface pressure contours from two computations, with propellers off and propellers on, is shown at a Mach number of 0.3 and an incidence of two degrees; the block boundaries of the surface blocks appear as lines in the figure. The pressure distribution on the wing and also on the horizontal tail surface is modified due to the presence of the propeller. The total pressure contours in the vertical plane of symmetry of the nacelle indicate the total pressure is practically constant in the entire flow field except in the region downstream of the actuator disk. The actuator disk model allows the assessment of slipstream-induced effects on the aerodynamic performance, stability, and control for general aerodynamic configurations, without the expense of computing the details of the rotating propeller flow.

4.5.6 Turbofan Nacelle Analysis System

A three-dimensional turbofan nacelle design system based on CFD has been in use at General Electric Aircraft Engines for several years.[23] The system was created to assist nacelle designers in the efficient assessment, modification, and improvement of design concepts. The grid generation, flow solution, and post processing are highly integrated in the system and tailored to the design applications of interest; the improved capabilities have reduced the design cycle time for the nacelle design process. While the code has the capability to model viscous effects by including the Reynolds-averaged Navier-Stokes terms, the Euler equations are generally solved in order to reduce the computational time. The engine is modeled as an actuator disk and the specification of the mass flow is equivalent to setting the lift coefficient for an external flow application, so that Euler computations would be expected to be very accurate outside the areas where significant viscous separation occur. An example of the validation studies which have been conducted for extensive applications at design and off-design conditions is shown in Fig. 4.5.6; the schematic of the nacelle geometry and the computational grid illustrate the multiblock structured-grid approach. The operating conditions for the computation corresponds to a typical cruise condition (M_∞ = 0.82, MFR = 0.65, α = 0 deg). The parameter MFR is the ratio of the captured free stream tube area to the inlet area and is representative of the engine mass flow ratio. The ideal Mach number is a commonly used design parameter and is computed using the local surface pressure and the free stream total pressure value. The longitudinal variation of the ideal Mach number indicates excellent agreement with the experiment. The entire computational time for a case is 30 minutes: 5 minutes for grid generation, 15 minutes for flow solution, and 10 minutes for post processing and initial design evaluation. Extensive computations are summarized in Ref. 23 for more than fifty cases, representing cruise, off-design, and take-off conditions, which have produced similar levels of agreement with experimental data.

4.5.7 Inlets

From the designer's point of view, an intake design delivers a specified mass flow with specified flow conditions at the engine face. An optimal design would provide these flow conditions with losses as low as possible. In particular, for hypersonic airbreathing vehicles, the specific intake design can be governed by the overall performance of the vehicle (i.e., it may no longer be possible to select an intake that is optimized by itself).

Therefore, flow simulation is an attractive tool to support intake design by providing detailed information on the flow structure which is necessary for the shape optimization process of all configurational elements and which is normally not provided by wind tunnel testing. Also, the scaling of wind tunnel data to realistic Mach and Reynolds numbers can be carried out by the help of flow simulation information, thereby lowering considerably uncertainty margins.

The application of Euler methods for intake design may be of high interest for all those cases where viscous effects are expected to be small. Another aspect[24,25] is the question as to whether the underlying inviscid approximation scheme of a Navier-Stokes method is able to provide Euler solutions with a minimum of numerical dissipation.

Although the calculations of scramjet flows presented in Ref. 25 demonstrated quite clearly the capability of the EUFLEX code to cope with flow phenomena in hypersonic intakes, these calculations also showed room for enhancements. This experience and insight led to further improvements of corresponding flux and limiter formulations. The benefits of these efforts are demonstrated by calculations for realistic two-dimensional airbreathing engine (turbo or RAM) intakes. The type of grid system used for corresponding computations is shown in Fig. 4.5.7. The grid depicted is called the fine mesh and consists of 301 grid points in x-direction and 107 grid points in z-direction. There are 52 cells spanning the height of the intake duct. The length of the intake duct behind the throat is somewhat shorter for the fine grid than for the medium grid (161x65 points) which was also considered. Some crude shock fitting has been attempted by arranging the surface grid points such that the point of the leading edge of the third ramp corresponds to the point on the tip of the intake lip.

The Mach number distribution for the Euler calculations on the fine grid is shown in Fig. 4.5.7. This finer grid produces sharper shocks in comparison to the coarser mesh; the third ramp shock for which the grid is adapted best is extremely thin in the fine grid solution. However, as a result of this "shock fitting" approach, the shock originating at the intake lip appears to be thicker than expected. The distribution of the mass flow deficit has been improved by the increase in grid points. From the computation, it is evident that the external ramp shocks merge outside the intake just below and behind the tip of the intake lip. On the outside of the intake lip, an expansion wave is generated which interacts with the slip stream behind the intersection point of the external ramp shocks and the shock formed by coalescence of all these shocks. Additional air-

intake computations for aircraft at transonic and supersonic speeds have been made by Buers et al.[26] and Grashoff.[27]

4.5.8 Scramjets

In 1990, Eberle et al.[25] made improvements to an earlier Euler scheme with the intent to enable the accurate and robust treatment of hypersonic flows. Emphasis was on improved capabilities of the characteristic-based method to capture shocks of any strength, to represent lee-side flows, and to represent base flows past vehicles cruising at any speed, particularly for hypersonic flow applications. An application of this generalized method is presented in Fig. 4.5.8, where high resolution results are shown for a Mach 3 scramjet problem quite often used to demonstrate inviscid code performance. The complex flow pattern exhibiting shocks, expansion fans, and slip lines is well resolved as can be deduced from the Mach contour and density contour plots. The numerical approach, which is based on a sophisticated flux formulation, may be viewed as an alternative to heuristic tools such as local grid enrichment or similar approaches. Although the shocks are far away from being aligned with grid lines, the capturing property of the characteristic-based method is good. It should be pointed out that the axial massflux error monitored is very small, and for most portions is well below one permille. Other calculations for nozzle flow fields are reported by Reidelbauch and Weiland.[28]

4.5.9 Nozzles

Euler codes intended for nozzle simulations at high Mach numbers require a high degree of numerical robustness. The EULFLEX algorithm includes an accurate Riemann solver, total temperature-preserving split flux vectors, as well as differentiable switches and interpolations. For preserving pressure/density positiveness at hypersonic speeds, an efficient implicit update procedure is proposed. Specific matrix preconditioning techniques are introduced to circumvent singularity effects of the associate Jacobian matrices. An application[29] in which all these items have proved beneficial is the inviscid flow simulation past a two–dimensional twin jet nozzle (Fig. 4.5.9). This configuration is a candidate design for a hypersonic aircraft. Computations are shown for the nozzle operating over a range of Mach numbers. The lower nozzle is that of a turbine, whereas the upper nozzle corresponds to a ramjet exhaust. At high Mach numbers (M_∞ = 6.8), only the ramjet is running. At medium supersonic Mach numbers (M_∞ = 3.5), both engines are in operation; at transonic speeds (M_∞ = 1.2), the ramjet is off and acts as a boundary layer ejector. In each case, the Mach number isoplots reveal extraordinarily sharp resolution of the flow structure.

4.5.10 References

1. KROLL, N., "Computations of the Flow Fields of Propellers and Hovering Rotors Using Euler Equations," Proc. of 12th European Rotorcraft Forum, Paper 28, 1986.

2. KROLL, N., "Berechnung von Stromungsfeldern um Propeller und Rotoren im Schwebeflug durch die Losung der Euler-Gleichungen," DLR-FB 89–37, 1989.

3. CARADONNA, F. X. and TUNG, C., "Experimental and Analytical Studies of a Model Helicopter Rotor in Hover," NASA TM–81232, 1981.

4. STAHL, H., "Application of a 3–D Euler Code to Transonic Blade Tip Flow," Proc. of the 12th European Rotorcraft Forum, Paper 29, 1986.

5. KRAMER, E. , HERTEL, J. and WAGNER, S., "Euler Procedure for Calculation of the Steady Rotor Flow with Emphasis on Wake Evolution," AIAA 90–3007–CP, Aug. 1990.

6. MacCORMACK, R. W., "The Effect of Viscosity in Hypervelocity Impact Cratering," AIAA 69–354, 1969.

7. KAU, H.-P. and GALLUS H. E., "Numerical Investigation of Unsteady Flow in Oscillating Turbine and Compressor Cascades," AGARD-CP-468 , Paper No. 2, 1989.

8. BENETSCHIK, H. and GALLUS, H. E., "Inviscid and Viscous Transonic Flows in Cascades Using an Implicit Upwind Algorithm," J. Propulsion and Power, Vol. 8, No. 2, 1992.

9. BOLCS, A. and FRANSSON, T. H. (eds.), "Aeroelasticity in Turbomachines — Comparison of Theoretical and Experimental Cascades Results," Communication du Laboratoire de Thermique Appliquee et de Turbomachines, No. 13, Lausanne, EPFL, 1986.

10. SAXER, A., FELICI, H., NEURY, C. and RYHMING, I. L., "Euler Flows in Hydraulic Turbines and Ducts Related to Boundary Condition Formulations," Proc. Seventh GAMM Conference on Numerical Methods in Fluid Mechanics, Vol. 20, Vieweg, 1988, pp. 343–354.

11. HAPPEL, H. W. and STUBERT, B., "Computation of Transonic 3-D Cascade Flow and Comparison with Experiments," AGARD CP-437, Paper No. 31, 1988.

12. HAPPEL, H. W., DIETRICHS, H. J. and LEHMANN, K., "Computation of Transonic 2D Cascade Flow and Comparison with Experiments," AGARD CP-401, Paper No. 19, 1986.

13. LECHELER, S. and FRÜHAUF, H. H., "A Fully Implicit 3–D Euler Solver for Accurate and Fast Turbomachinery Flow Calculation," Proc. Ninth GAMM Conference on Numerical Methods in Fluid Mechanics, Lausanne, Switzerland, Sept. 25–27, 1991.

14. LEICHER, S., "Numerical Simulation of Internal and External Inviscid and Viscous 3D Flows," AGARD CP-412, Paper No. 14, 1986.

15. WHITFIELD, D. L., SWAFFORD, T. W., JANUS, J. M., MULAC, R. A. and BELK, D. M., "Three-Dimensional Unsteady Euler Solutions for Propfans and Counter-Rotating Propfans in Transonic Flow," AIAA 87–1197, June 1987.

16. JANUS, J. M. and WHITFIELD, D. L., "A Simple Time-Accurate Turbomachinery Algorithm with Numerical Solutions of an Uneven Blade Count Configuration," AIAA 89–0206, Jan. 1989.

17. JANUS, J. M., "Advanced 3–D CFD Algorithm for Turbomachinery," Ph.D. Dissertation, Mississippi State University, May 1989.

18. JANUS, J. M. and WHITFIELD, D. L., "Counter-Rotating Propfan Simulations Which Feature a Relative-Motion Multi-block Grid Decomposition Enabling Arbitrary Time Steps," AIAA 90–0687, Jan. 1990.

19. JANUS, J. M., HORSTMAN, H. Z. and WHITFIELD, D. L., "Unsteady Flowfield Simulation of Ducted Prop-Fan Configurations," AIAA 92–0521, Jan. 1992.

20. STEFFEN, F. W., "Cruise Performance of an Isolated 1.15 Pressure Ratio Turbofan Propulsion Simulator at Mach Numbers from 0.6 to 0.85," NASA TM X-3064, 1974.

21. BOCCI, A., BOUSQUET, J. M. and LOHMANN, D., "Euler Calculations for Single-Rotation High Speed Propfan: Part III," GARTEUR TP059, ARA-Report, Dec. 1990.

22. BOERSTOEL, J. W., "Conceptual Design of a Computer-Code System for the Calculation of Flows Around Transport Aircraft," NLR TP 92065 L, Sept. 1991.

23. UENISHI, K., PEARSON, M. S., LEHNIG, T. R. and LEON, R. M., "CFD-Based 3D Turbofan Nacelle Design System," AIAA 90-3081, Aug. 1990.

24. BISSINGER, N. C. and EBERLE, A. "CFD Contributions During Hypersonic Airplane Intake Design," AGARD-CP 510, paper 26, 1991.

25. EBERLE, A., SCHMATZ, M. A. and BISSINGER, N. C., "Generalized Flux Vectors for Hypersonic Shock Capturing," AIAA 90–0390, 1990.

26. BUERS, H., LEICHER, S. and MACKRODT, P. A., "Numerical and Experimental Investigation of Engine Inlet Flow with the Dornier EM2 Supersonic Inlet Model," AGARD CP, Paper No. 32, 1988.

27. GRASHOF, J. and SCHMIDT, W., "Transonic Intake Flow," ICAS 82–4.3.1, 1982.

28. RIEDELBAUCH, C. and WEILAND, C., "Inviscid Laval-Nozzle Flowfield Calculation," J. Spacecraft, Vol. 25, 1988, pp. 88–96.

29. EBERLE, A., "Characteristic Fluxvectors for High Speed Flow Solutions,"*Advances in Computational Fluid Dynamics*, (eds.: W. G. Habashi and M. M. Hafez), Pineridge Press, Swansea, 1992.

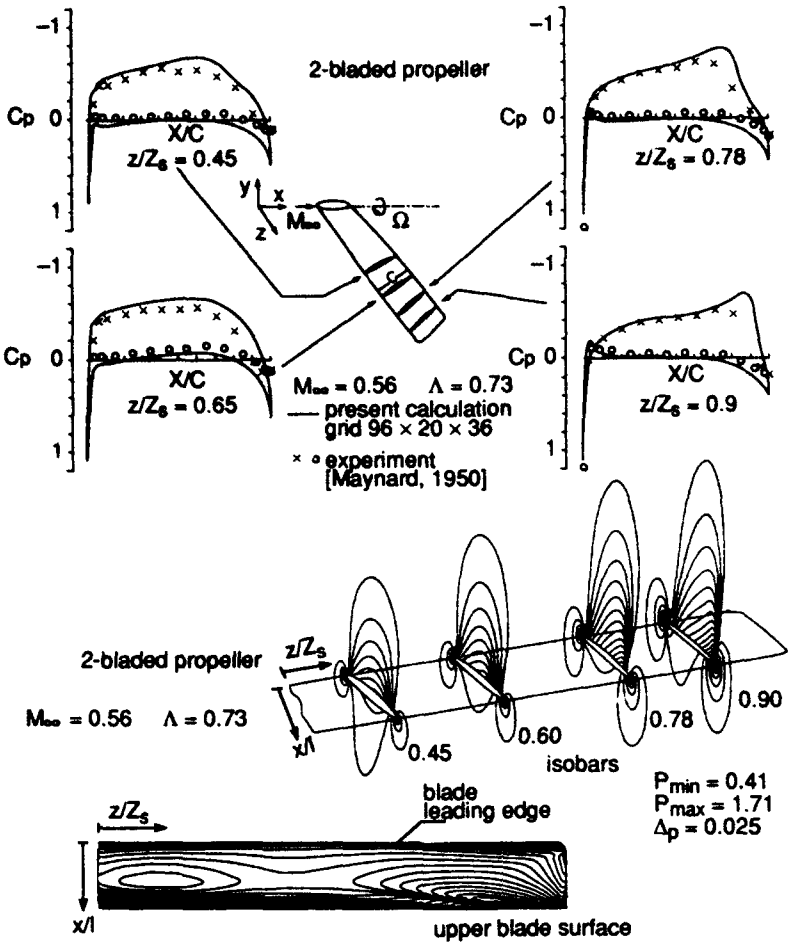


Figure 4.5.1 Euler calculation for propeller.

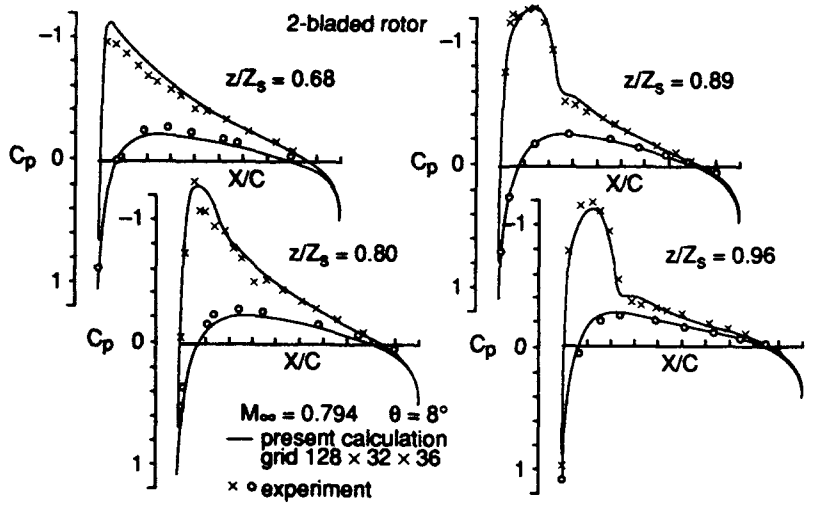
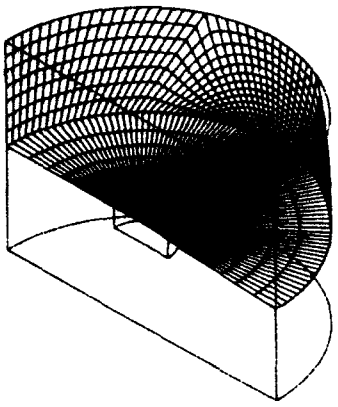
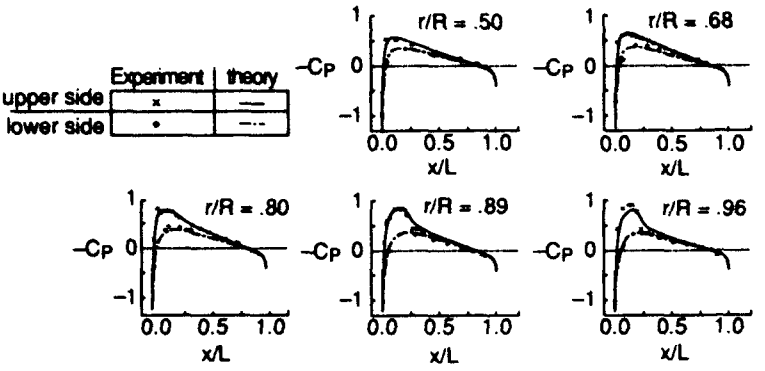


Figure 4.5.2 Euler calculation for hovering rotor.





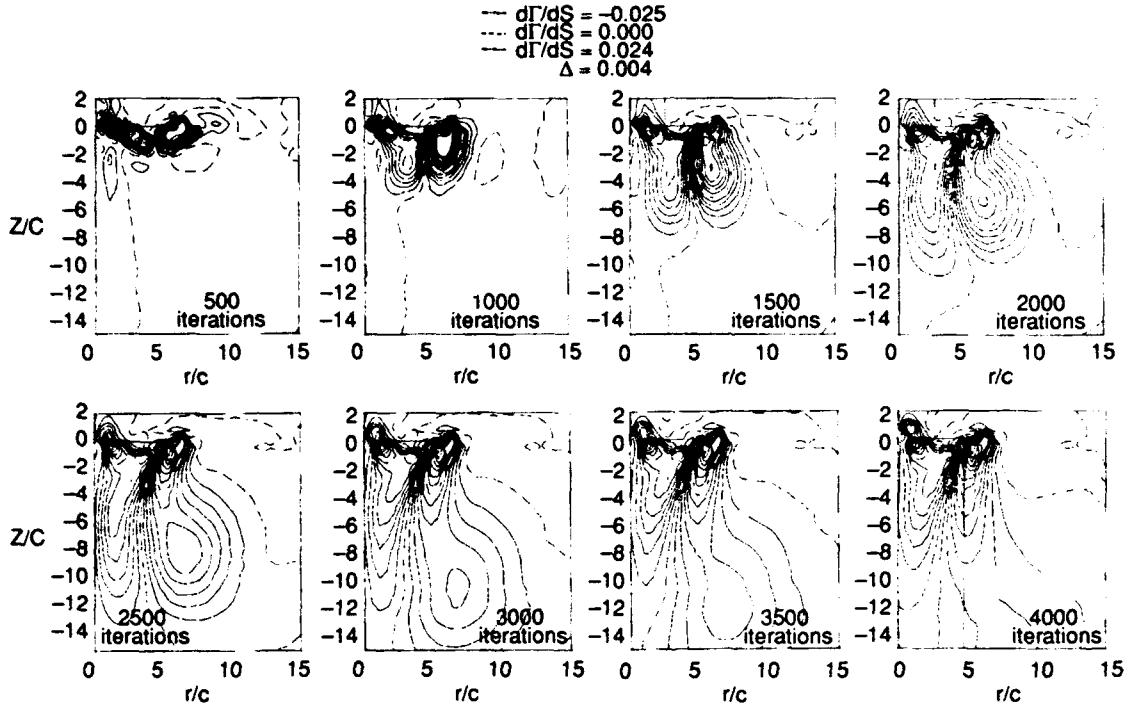
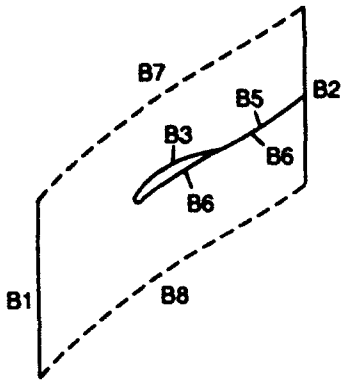
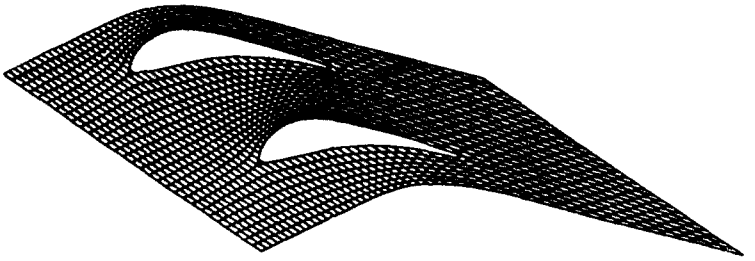
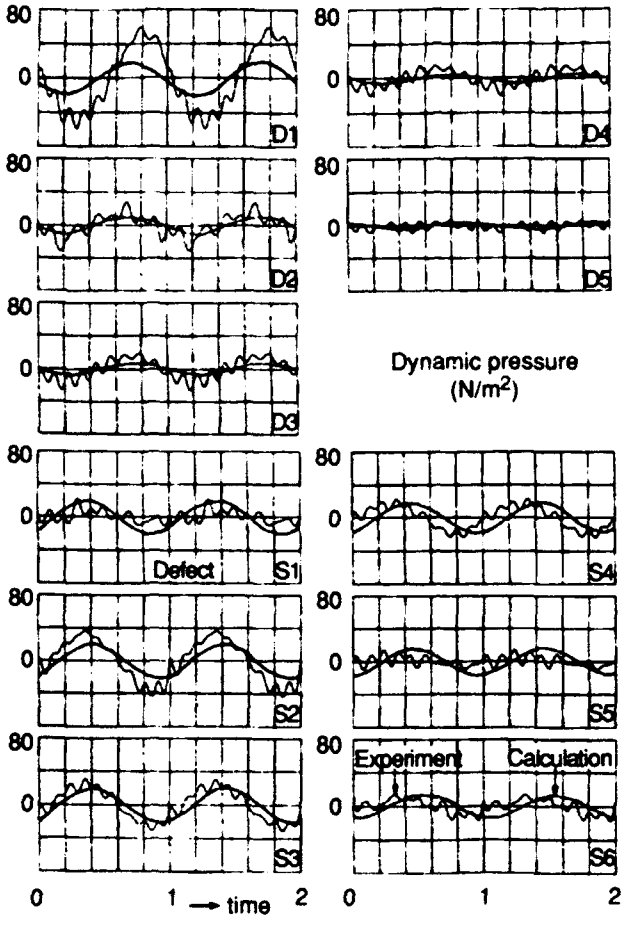


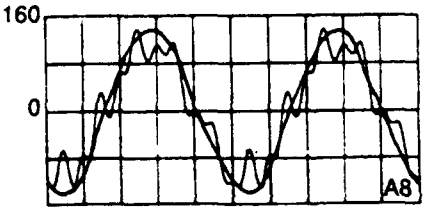
Figure 4.5.3 Euler calculations for hovering rotor with complete wake model showing grid, pressure distributions, and temporal evolution of starting process in terms of circulation per unit area contours.





B1 - Upstream boundary
B2 - Downstream boundary
B3, B4 - Blade surface
B5, B6 - Slipstream
B7, B8 - Periodical boundary





Measured acceleration (m/s²) in comparison with the calculation

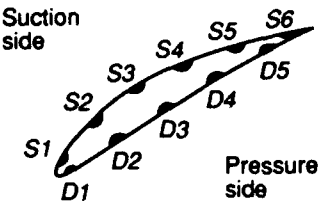
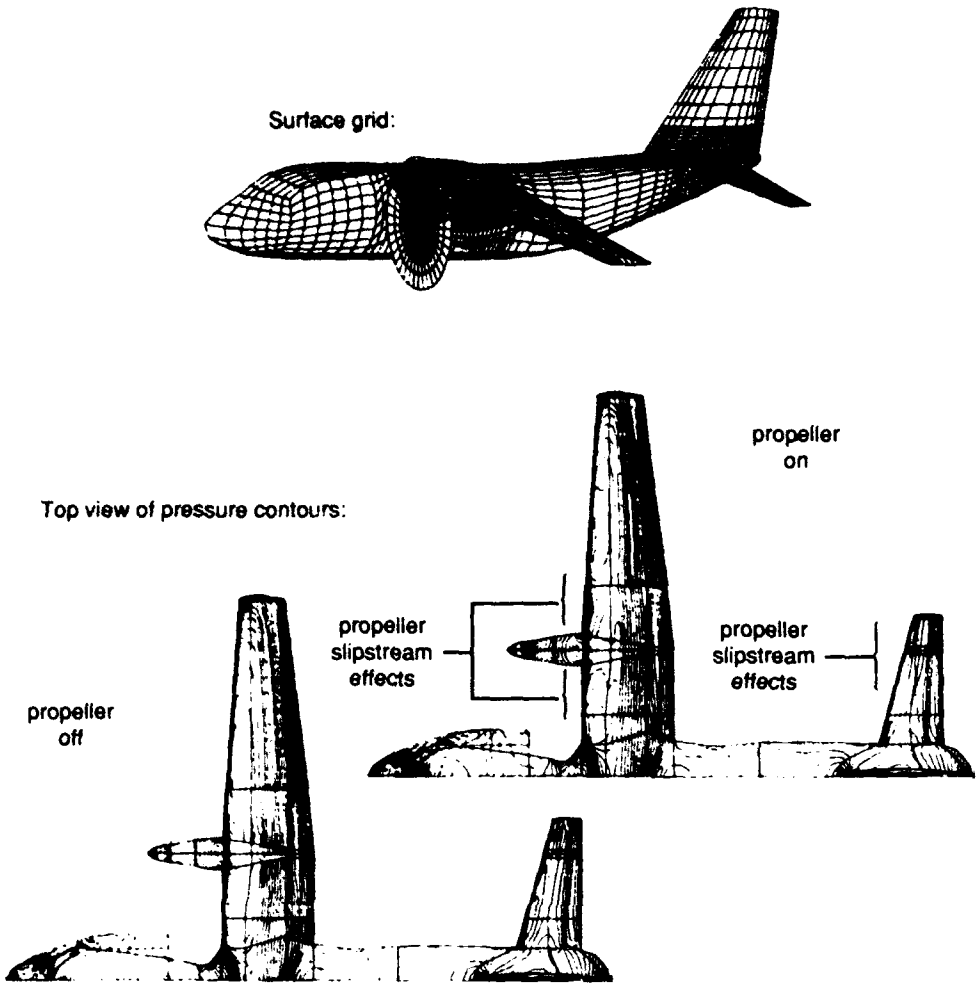


Figure 4.5.4 Euler computations for a turbine cascade showing grid and comparisons between calculation and experiment for cascade bending in TWM; σ = 180 deg, ν = 132 Hz, M_1 = 0.228.



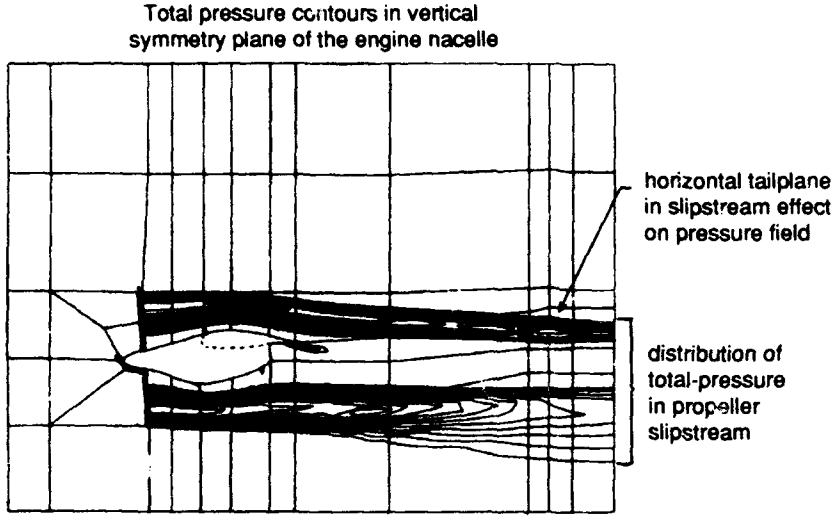
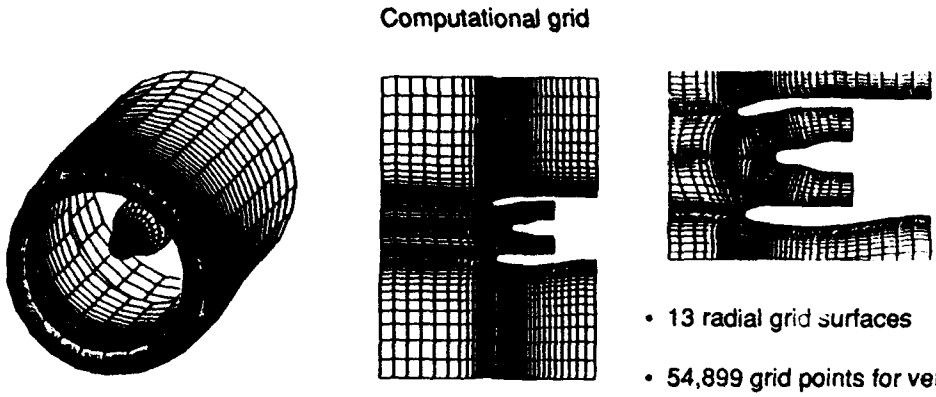


Figure 4.5.5 Actuator disk modeling of propeller slipstream effects for a high-wing subsonic transport aircraft.



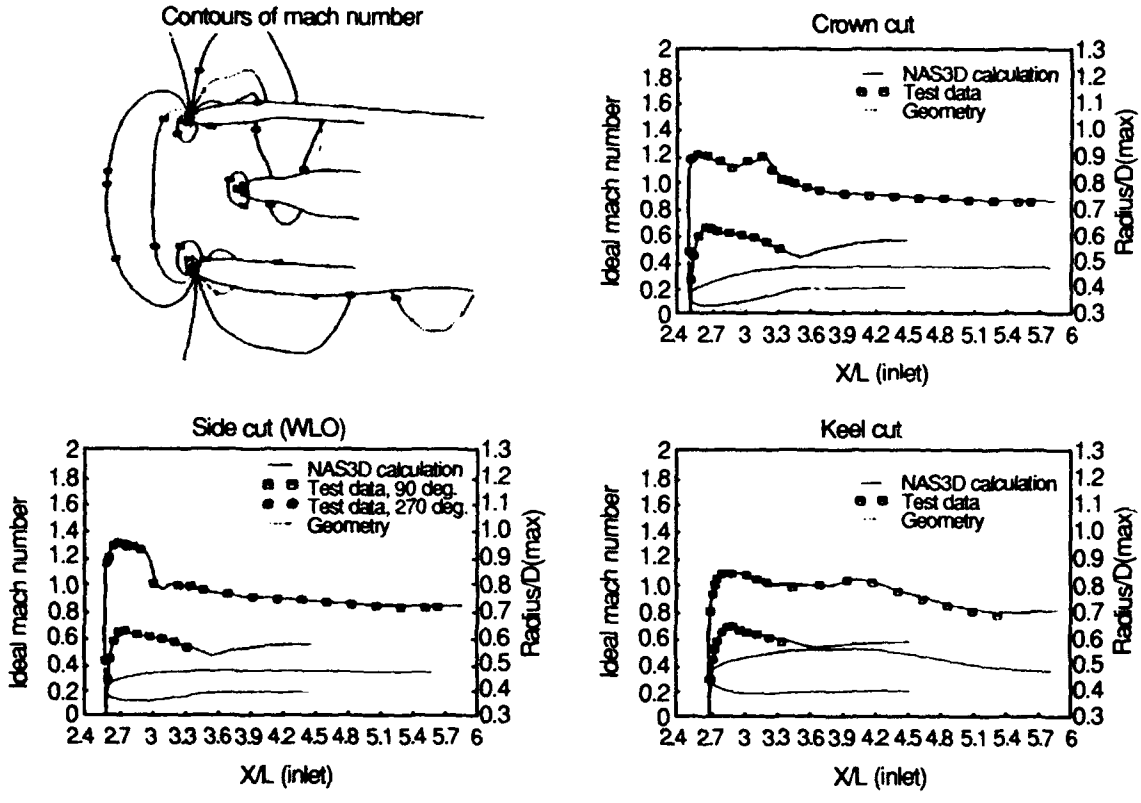


Figure 4.5.6 Turbofan nacelle analysis system.

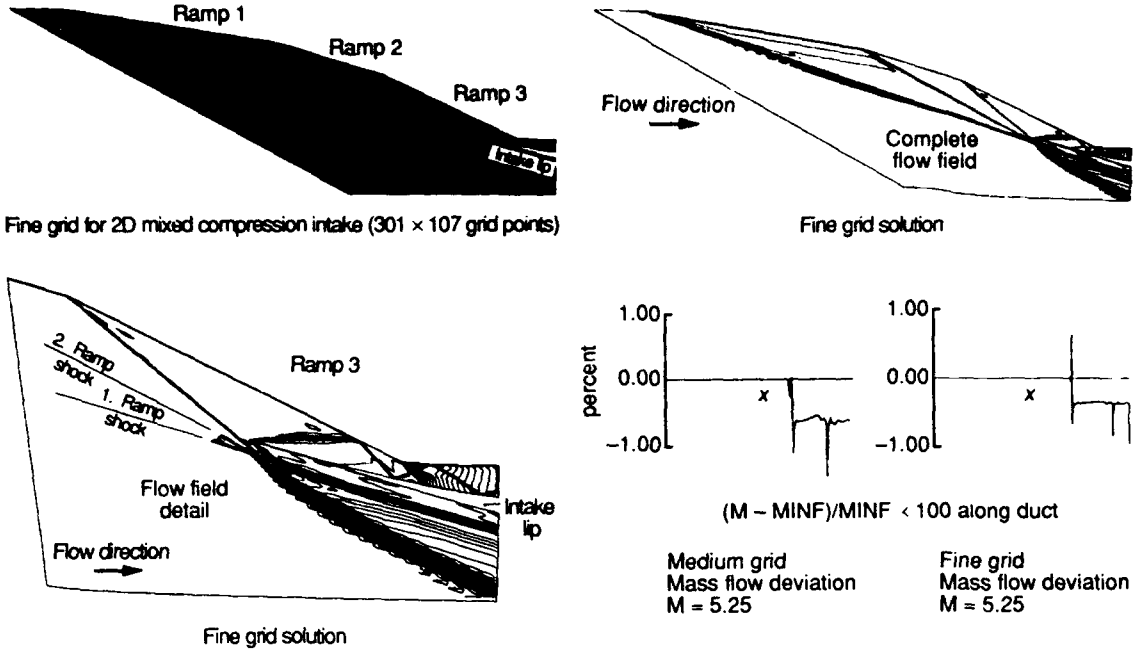


Figure 4.5.7 Fine grid, lines of constant Mach number, and mass flow axial variations for Euler computations of a mixed compression inlet; $M_\infty = 5.25$.

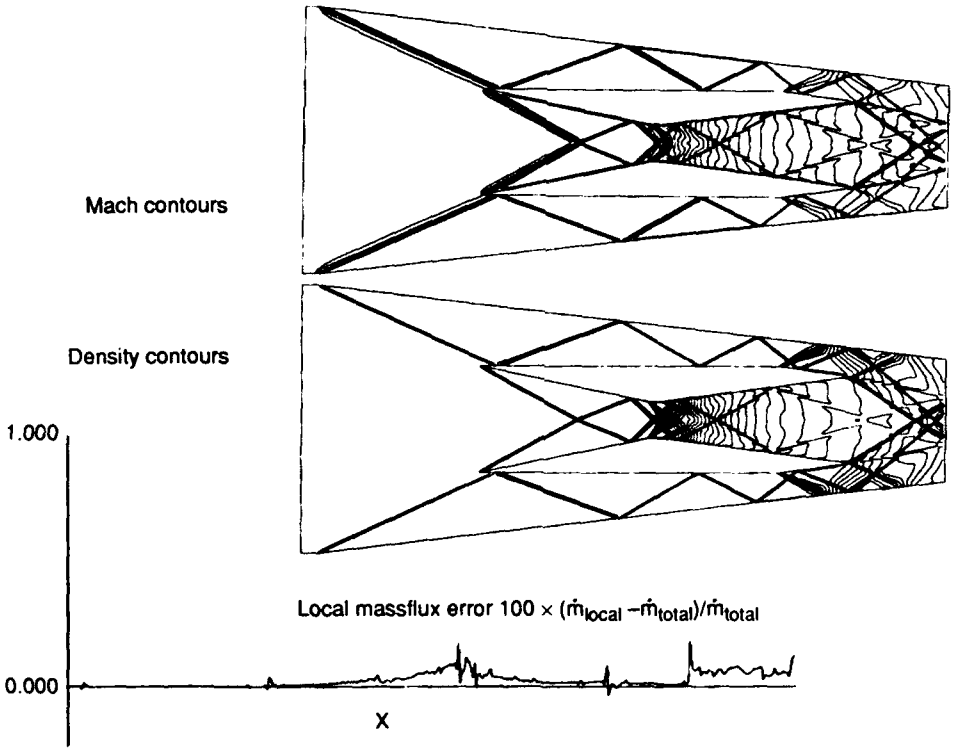
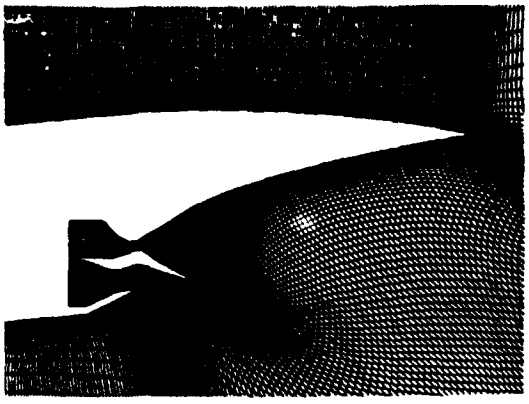
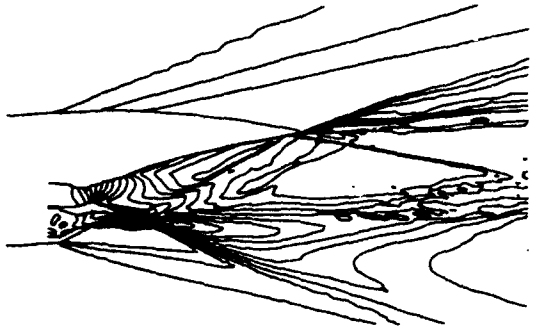


Figure 4.5.8 Mach contours, density contours, and local massflux errors from Euler computations of a scramjet flowfield.

Partial view of computational grid

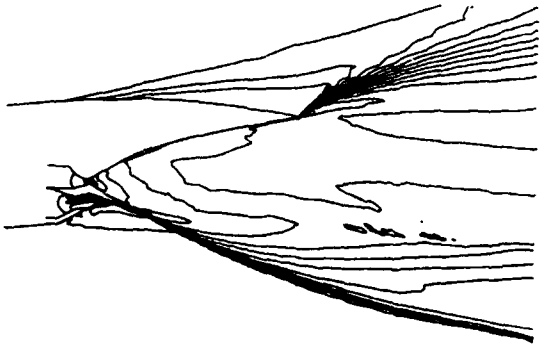




M = 3.500, a = 4.400°

Machnumber increment = .2

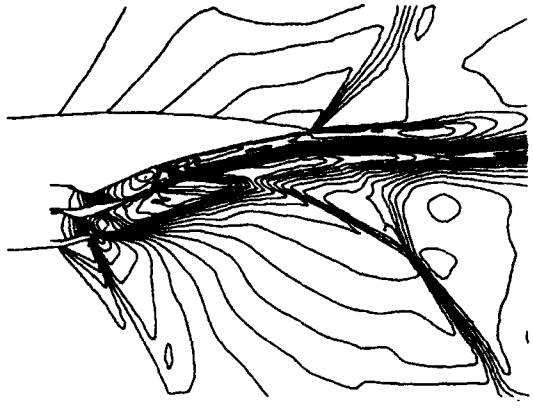
Pinfinity = 5579.00 Pascal
Tinfinity = 216.65 Kelvin
Ptotalramjet = 224000.00 Pascal
Ttotalramjet = 2555.00 Kelvin



M = 6.8, α = 6.0°

Mach number increment = .4

Pinfinity = 1030.00 Pascal
Ptotalramjet = 674000.00 Pascal
Ttotalramjet = 3086.00 Kelvin



M = 1.200, α = 4.400°

Mach number increment = .1

Pinfinity = 26440.00 Pascal
Tinfinity = 223.15 Kelvin
Ptotalramjet = 227000.00 Pascal
Ttotalramjet = 2372.00 Kelvin

Figure 4.5.9 Grid and Mach contours of a twin nozzle computation for a hypersonic aircraft at different operating conditions.

4.6 HYPERSONIC FLOWS

Because of recent renewed interest in the development of hypersonic air-breathing aircraft, such as the National Aero-Space Plane in the U.S., there has been active research and development of Euler solvers for hypersonic flows. The emphasis has been principally in two areas: the design of oscillation-free schemes capable of capturing strong shocks and the incorporation of gas models more sophisticated than a perfect gas. Both are central ingredients of an algorithm able to operate at high Mach numbers reliably that could form the framework for a hypersonic, viscous algorithm. Several examples of the improved shock resolution available through either adaptation or improved dissipation models are shown below for the case of a plane shock intersecting a blunt body at hypersonic speeds, corresponding to a shock-shock interaction problem. The hypersonic flow over a double ellipsoid body was computed extensively with different Euler solvers at the INRIA Workshop on Hypersonic Flows for Reentry Problems;[1] two examples are shown: a perfect gas and a real gas computation. The proceedings of the workshop provide a summary of the methodology which currently exists for applications to hypersonic flows. An application of Euler solvers to the equilibrium and nonequilibrium flow over a Hermes configuration is also shown.

4.6.1 Blunt Cylinders

Algorithmic details and general capabilities for the simulation of inviscid hypersonic flows in chemical non-equilibrium conditions are given by Pfitzner.[2,3] The flow solver is based on a quasi-conservative split matrix method with upwind-biased space discretization coupled to a Runge-Kutta time-stepping scheme. The chemistry source terms are treated either explicitly or (point) implicitly. The chemical kinetics are based on the 5-species 17-reaction model according to Park.[4] For simulation of nonequilibrium reentry air flows at heights above 50 km, explicit treatment of source terms was sufficient.

Figure 4.6.1 shows a comparison of fringe patterns for inflow of partially dissociated nitrogen about a 2–inch diameter cylinder at the following flow conditions: $M_\infty = 6.14$, $p_\infty = 2910\ Pa$, $T_\infty = 1833K$, $u_\infty = 5590m/s$, $c_\infty^N = 0.073$. Corresponding experiments were conducted by Hornung.[5] Whereas the shock position is reproduced well, the fringe patterns differ somewhat due to the one-temperature model used which does not allow for an appropriate delay of chemical reactions by thermal nonequilibrium in the vibrational degrees of freedom.

Figure 4.6.1 displays a comparison of temperature contours of a flow about a 1/4 inch sphere in air at free stream conditions of: $M_\infty = 15.3$, $p_\infty = 664\ Pa$, $T_\infty = 293\ \deg K$, $u_\infty = 5280m/s$. The results illustrate the effects of ideal, equilibrium and chemical nonequilibrium real gas assumptions. Also shown (top right of Fig. 4.6.1) is a comparison of the resulting shock contours with the experiment of Lobb.[6] The experimental shock standoff distance is slightly larger than the calculated one due to thermal non-equilibrium effects.

4.6.2 Shock-Shock Interactions

Adaption Effects

The adaptive-grid method can be used to resolve multiple-shock interactions computed with shock-capturing methods. An example (shown in Fig. 4.6.2) uses an unstructured data management scheme allowing flexible grid adaptation with local refinement in one or two coordinate directions. Specialized adaption criteria account for the expected phenomena in super- and hypersonic flows, for example, strong shocks and slip lines. The Euler equations are solved using a second order upwind discretization[7,8] according to Harten and Yee[9]. The scheme is a finite volume method based on quadrilateral meshes.

Fig. 4.6.2 presents a complex example which permits an evaluation of the effectiveness of unidirectional cell division.[8] This study uses a Mach 8.03 shock-shock interaction[7] as a test case. This example is well suited for comparisons of methods since the flow field complexity stems from the gas dynamic interaction and not from some arbitrarily chosen boundary. Additionally, the solution demands resolution of many different and disparate convective length scales and contains both sub- and supersonic regions.

Taking the origin at the center of the cylinder, the problem is completely specified by a Mach 8.03 free stream containing an impinging shock which follows a prescribed line (y=0.3271 x + 0.41471). Figure 4.6.2 contains two discrete adapted-grid solutions to this problem (each was converged with a constant global time step for 25.0 characteristic times). The top solution did not make use of unidirectional cell division, and the mesh shown contains 12869 nodes. The Mach contours displayed maintain an increment of 0.25 and the Mach 1 line is marked. The bottom solution used directional adaptation at the finest level, and resulted in 8510 nodes. Inspection of this case reveals that the bow shock, the slip lines that bound the supersonic jet, and the supersonic region near the upper portion of the cylinder are adapted unidirectionally. Prior to the final division sequence, both meshes were identical and contained 5500 nodes.

Upwind Discretization Effects

The effect of upwind discretization is shown for this same test case in Fig. 4.6.3; a sketch of the flow field is also shown. The CEVCAT code was modified and extended for the calculation of super- and hypersonic flows, which are characterized by strong nonlinearities, like shocks, slip lines, and shock-shock interactions. The central spatial discretization of the convective fluxes has been better adapted to the artificial dissipative operator.[10] In connection with a special boundary treatment of the discretization at the walls, the robustness of the central method was significantly improved and solutions with strong shocks at high Mach numbers and high angles of attack were computed. For a better resolution of discontinuities, the upwind TVD discretization according to Harten and Yee[9] was implemented.[11] Figure 4.6.3 shows the comparison between a central and an upwind weighted discretization.[12] In contrast to the central scheme, the upwind scheme resolves shocks within 2 grid cells and gives a good prediction of the complex flow with shock-shock interaction. Further comparisons between schemes are compared systematically by Kroll et al.[12] for this shock-shock interaction.

4.6.3 Double Ellipsoid

Perfect Gas

The flow over a double ellipsoid shape (representative of the

forward section of a hypersonic vehicle), computed with an adaptive unstructured-grid method,[13,14] is shown in Fig. 4.6.4. The conditions correspond to M_∞ = 8.15 and α = 30 deg; the case is one of the test conditions defined for the Workshop on Hypersonic Flows for Re-Entry Problems[1] and has been extensively computed by a number of different contributors. The mesh and pressure contours are shown for the surface, the longitudinal plane of symmetry, and a lateral plane at the end of the body for an initial solution and an adapted mesh solution. The bow shock is positioned very close to the body at this Mach number and is resolved much more sharply with the adapted mesh, owing to the clustering of points near the body and a thinning of points outside the bow shock. The cross flow shock is similarly much better resolved.

Real Gas

At higher Mach numbers, the density distributions in the cross-flow and in the symmetry plane are qualitatively very similar between real gas and perfect gas computations. However, there are some major differences within the Mach number distribution, as shown in the computations by Dortmann[15] (Fig. 4.6.5), where the temperature distribution and the mole fraction distribution of molecular oxygen is given. The conditions correspond to M_∞ = 25 and α = 30 deg, corresponding to another of the conditions prescribed for the INRIA Workshop on Hypersonic Flows for Re-Entry Problems.[1] On the windward side, the complete oxygen dissociation takes place within or shortly behind the bow shock. On the leeward side, the temperature jump across the bow shock is not as strong and the dissociation occurs within a small layer which emanates from the bow shock near the nose and follows a plane of constant temperature, T=2500 deg K. This relatively rapid change within the gas mixture leads to a sudden change in the speed of sound, which influences the Mach number. The same effect due to nitrogen dissociation can be observed on the windward side following a plane of constant temperature with T=5000 deg K. However, the dissociation is not complete since the temperature level is not high enough.

4.6.4 Hermes

National and European projects supporting manned space activities have initiated over the last years considerable efforts aiming on the improvement of simulation tools as essential elements for a cost-effective design procedure for manned reentry vehicles. The pacing item of that development is the need to reduce uncertainty margins from the aerothermodynamic data set. Uncertainties can be reduced considerably if certain extrapolation rules relating experimental to free flight data are verified, as could be done with the use of validated flow simulation methods. Increased uncertainty results especially from the fact that for the most critical parts of a typical reentry trajectory, an application of simple similarity laws does not allow one to relate ground-based measurements to free-flight data.

Those numerical methods which could contribute to the simulation of relevant fluid mechanical phenomena during reentry have been systematically developed the last years. One of the Euler codes enabling three-dimensional inviscid flow simulation under equilibrium and nonequilibrium conditions was developed by Weiland and Pfitzner[3,16] and applied by Hartmann and Weiland[17] to a configuration of the Hermes reentry verhicle. The code is built around a quasi-conservative splitmatrix formulation relying on a shock-fitting technique for the outermost bow shock and a capturing approach for all other embedded shock waves.

The configuration with deflected elevon flaps in different perspective views is shown in Fig. 4.6.6. The surface mesh reveals a different mesh structure for the forebody and aft fuselage region indicating a mesh refinement in the circumferential direction to improve the flow resolution past the fuselage in the region of the main wing and winglet sections. Because the shock-fitting approach used mesh adaptation capabilities, the volume mesh is divided into two blocks (as displayed for the last mesh cross-section). The inner mesh block is fixed in time whereas the outer boundary of the second mesh block can be adjusted to the bow shock shape according to the Rankine-Hugoniot conditions.

Euler solutions are presented for equilibrium and chemical nonequilibrium real gas conditions for the following flow conditions: M_∞ = 25, α = 30 deg, β = 0 deg. Fig. 4.6.6 shows corresponding Mach number distributions on the vehicle surface. Contour lines are clustered on the leeside wing and the upper fuselage, as well as in front of the canopy section. Whereas the canopy region is exposed to a recompression shock due to the ramp effect of the forebody geometry, on the leeside of the wing a strong expansion around the round wing leading edge takes place. This expansion leads to a cross flow recompression shock near the wing/fuselage intersection. The same effect is responsible for the Mach contour clustering on the upper fuselage near the symmetry plane. Although the same free-flight conditions are used in both computations, it is interesting to note that under the assumption of chemical nonequilibrium, the footprint of the leeside wing cross flow shock is changed dramatically and also affects the winglet sections.

Whereas inviscid flow solutions show interesting gas dynamic effects one, viscous interaction effects can alter any conclusions drawn from inviscid flow results. However, for determining the bandwidth of results for global forces and moments and the identification of effects due to basic physical assumptions, Euler simulations are quite valuable for direct design support.

4.6.5 References

1. DESIDERI J.-A., et al. (ed.), *Workshop on Hypersonic Flows for Reentry Problems*, Vols. 1–2, Springer Verlag, 1991.

2. PFITZNER, M., "Simulations of Inviscid Equilibrium and Non-Equilibrium Hypersonic Flows, Proceedings of the 12th Int. Conference on Numerical Methods in Fluid Dynamics," *Lecture Notes in Physics*, Vol. 371, pp. 432–436, Springer Verlag, 1990.

3. PFITZNER M., "A 3D Non-Equilibrium Shock-Fitting Algorithm Using Effective Rankine-Hugoniot Relations," AIAA 91–1467, 1991.

4. PARK C., "On the Convergence of Computation of Chemically Reacting Flows," AIAA 85–0247, 1985.

5. HORNUNG H. G., "Non-Equilibrium Dissociating Nitrogen Flow over Spheres and Circular Cylinders," J. Fluid Mechanics, Vol. 53, pp. 149–176 , 1972.

6. LOBB, R. K., "Experimental Measurement of Shock Detachment Distance on Spheres Fired in Air at Hypervelocities,"

The High Temperature Aspects of Hypersonic Flow, ed. W. C. Nelson, Pergamon Press (1964).

7. AFTOSMIS, M. J., "An Adaptive Grid Algorithm for Nonequilibrium Hypersonic Flows," M.S. Thesis, Massachusetts Institute of Technology, Cambridge, MA, 1989.

8. AFTOSMIS, M. J. and KROLL, N., "A Quadrilateral Based Second-Order TVD Method for Unstructured Adaptive Meshes," AIAA-91-0124, 1991.

9. YEE, H. C. and HARTEN, A., "Implicit TVD Schemes for Hyperbolic Conservation Laws in Curvilinear Coordinates," AIAA Journal, Vol. 25, pp. 266-274, 1987.

10. SCHONE, J., STREIT, T. and KROLL, N., "Steps Towards An Efficient And Accurate Method Solving The Euler Equations Around A Re-Entry Configuration At Super- And Hypersonic Speeds," Proceedings 1st European Symposium on Aerothermodynamics for Space Vehicles, Noordwijk, The Netherlands, 28-31 May 1991.

11. KROLL, N. and ROSSOW, C.-C., "A High Resolution Cell Vertex Scheme for the Solution of the Two- and Three-Dimensional Euler Equations," 12th International Conference on Numerical Methods in Fluid Dynamics, Oxford, GB, 1990.

12. KROLL, N., GAITONDI, D. and AFTOSMIS, M. J., "A Systematic Comparative Study of Several High Resolution Schemes for Complex Problems in High Speed Flows," AIAA 91-0636, Jan. 1991.

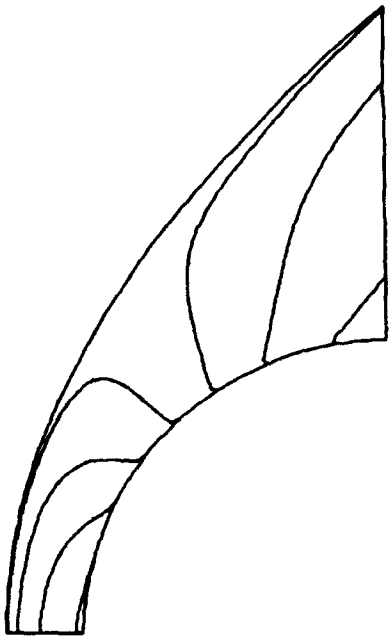
13. MORGAN, K., PERAIRE, J. and PEIRO, J., "Unstructured Grid Methods for Compressible Flows," AGARD Report 787, Special Course on Unstructured Grid Methods for Advection-Dominated Flows," 1992, pp. 5.1–5.39.

14. HASSAN, O., PEIRO, J., PERAIRE, J. and MORGAN, K., "The Application of an Adaptive Unstructured Grid Method for the Solution of Hypersonic Flows Past Double Ellipse and Double Ellipsoid Configurations," *Hypersonic Flows for Re-Entry Problems*, Springer-Verlag, 1992.

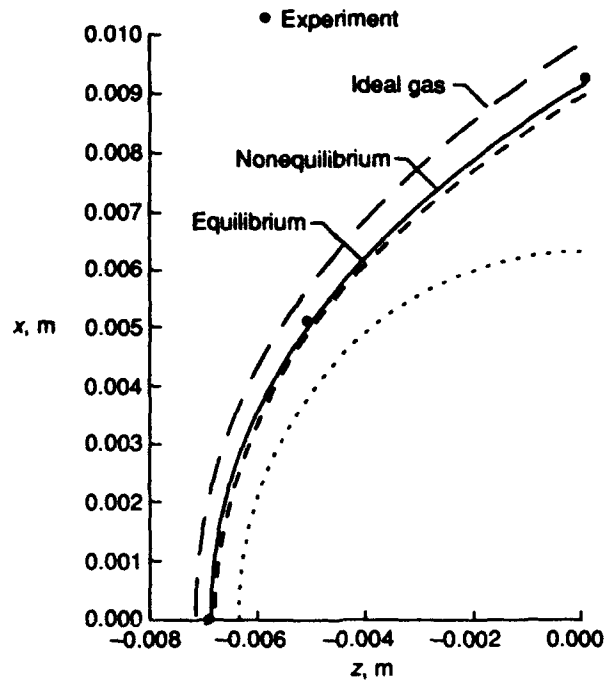
15. DORTMANN, K., "Viscous and Inviscid Hypersonic Flow About a Double Ellipsoid," *Workshop on Hypersonic Flows for Reentry Problems*, ed. J.-A. Desideri et al., Vol. 2, pp. 616–634, Springer Verlag, 1991.

16. PFITZNER M and WEILAND C., "3D Euler Solutions for Hypersonic Mach Numbers," Paper 22 of AGARD-CP 428, 1987.

17. HARTMANN, G. and WEILAND, C., "Stromungsfeldberechnungen um den HERMES Raumgleiter wahrend der Wiedereintrittsphase," DGLR-Jahrbuch, Vol. 1, Paper 91–223, 1991, pp. 255–264.



Fringe contours of $\rho/\rho_\infty(1 + 0.28c_N)$



T/T_∞ contours, $\Delta = 1$

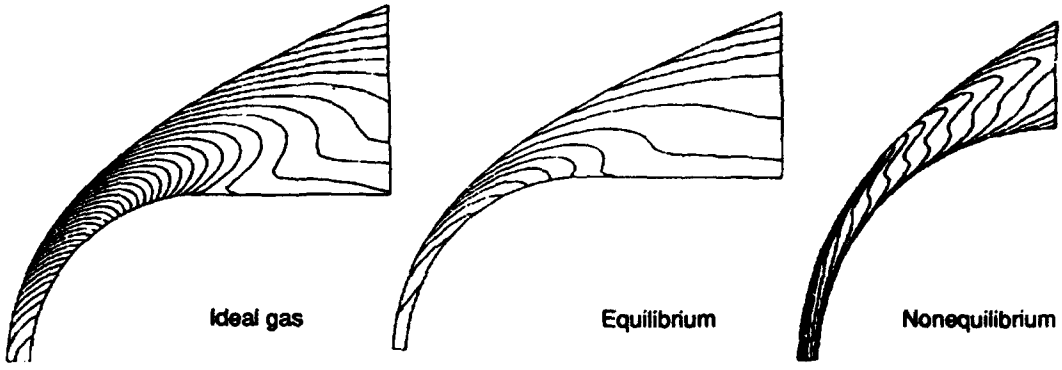


Figure 4.6.1 Blunt body ideal, equilibrium, and nonequilibrium gas Euler computations.

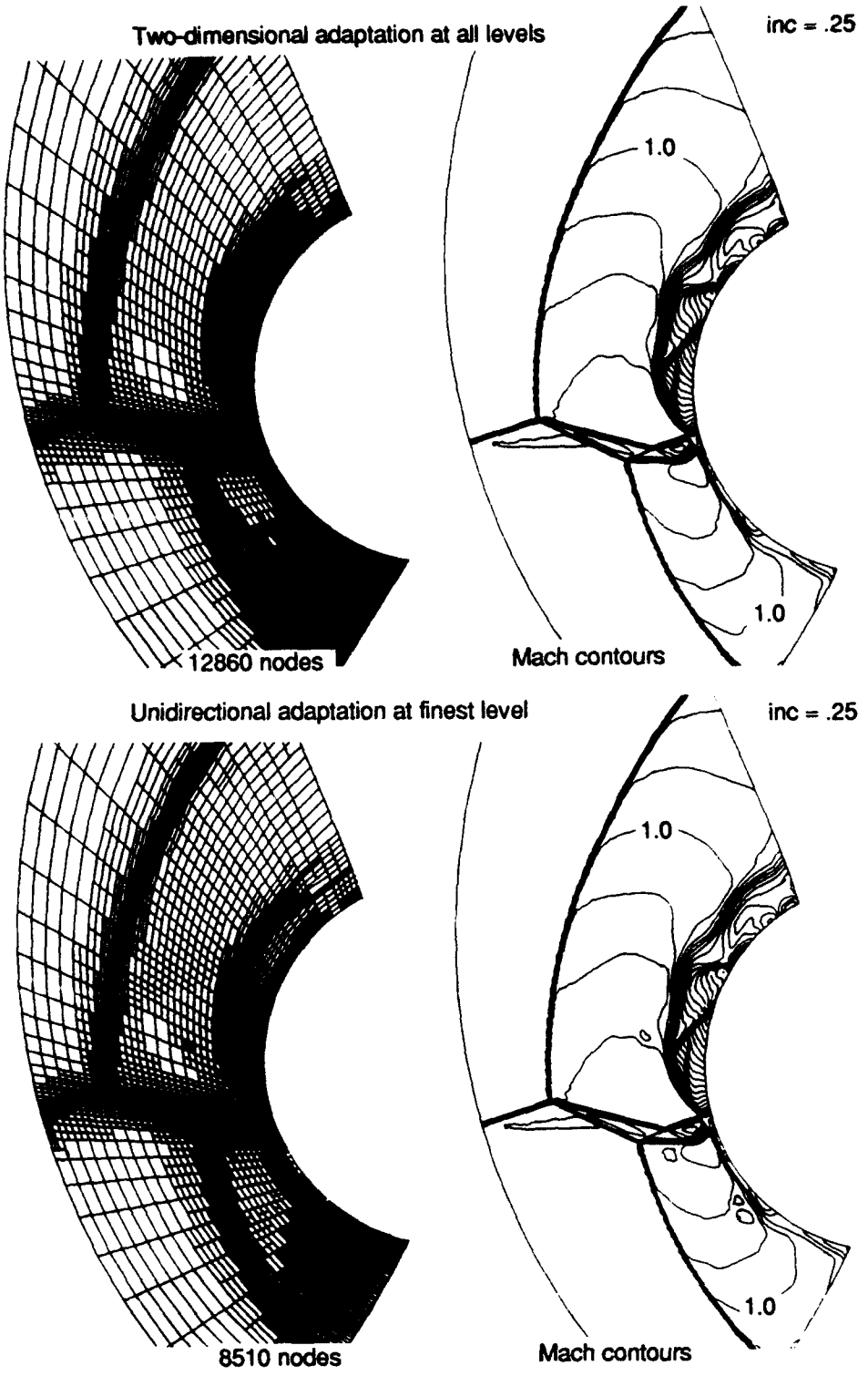
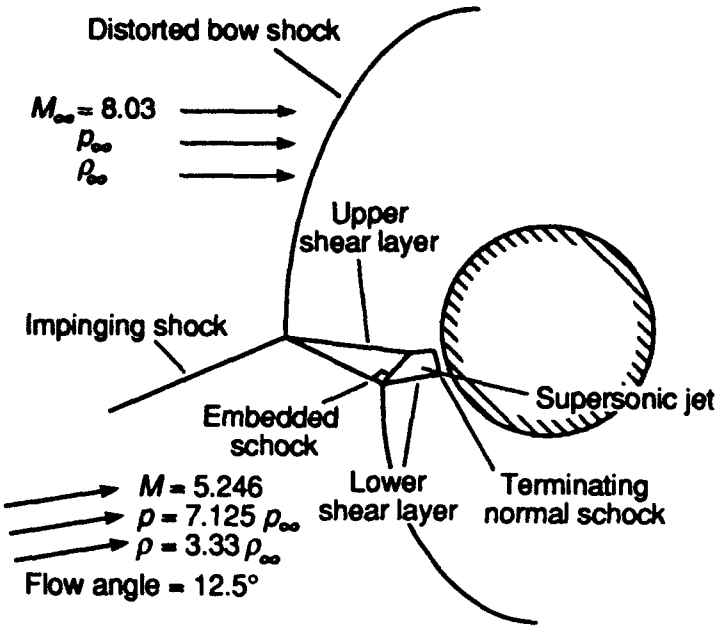


Figure 4.6.2 Mach contours for shock-shock interaction computed with (lower: 8510 nodes) and without (upper: 12,860 nodes) unidirectional adaptation applied at the finest adaptive level.



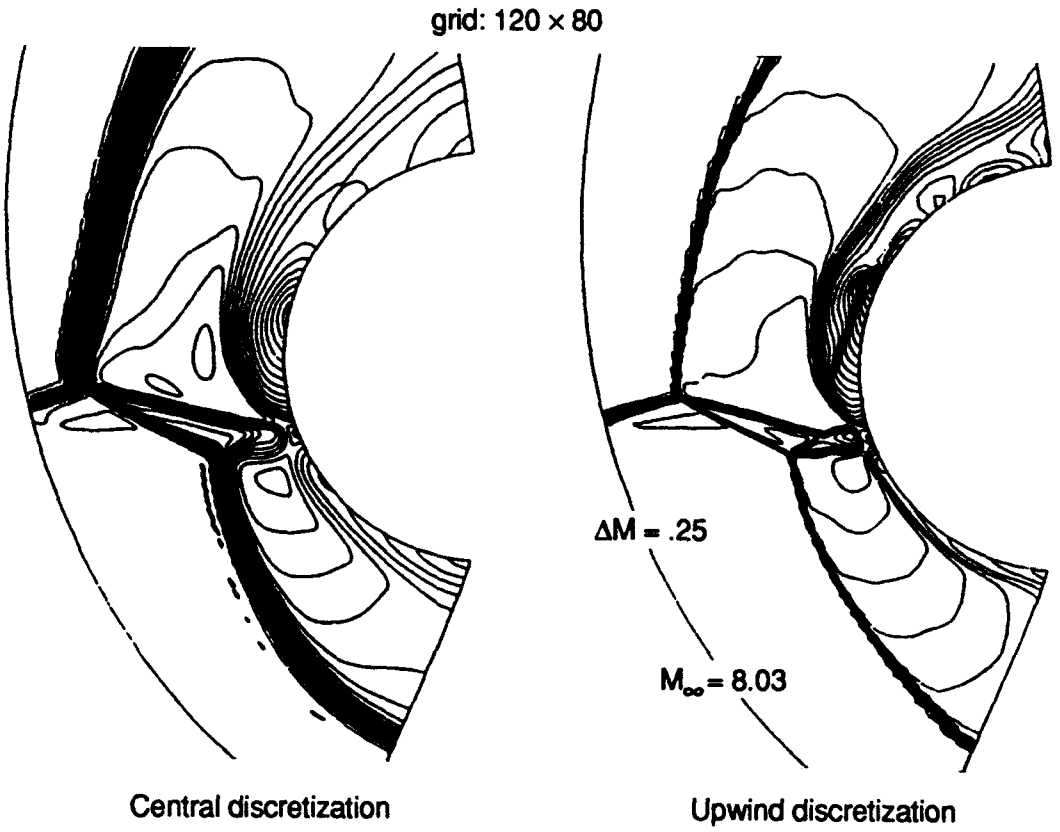
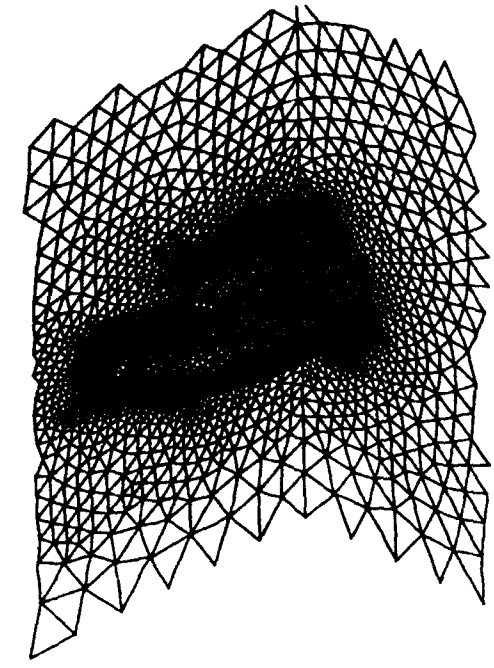
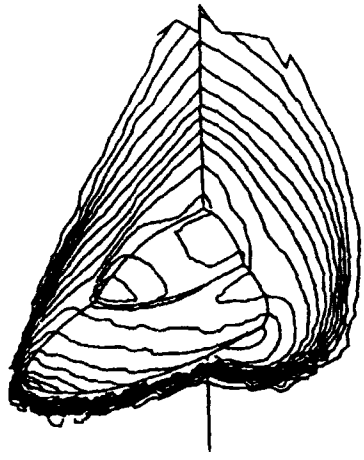
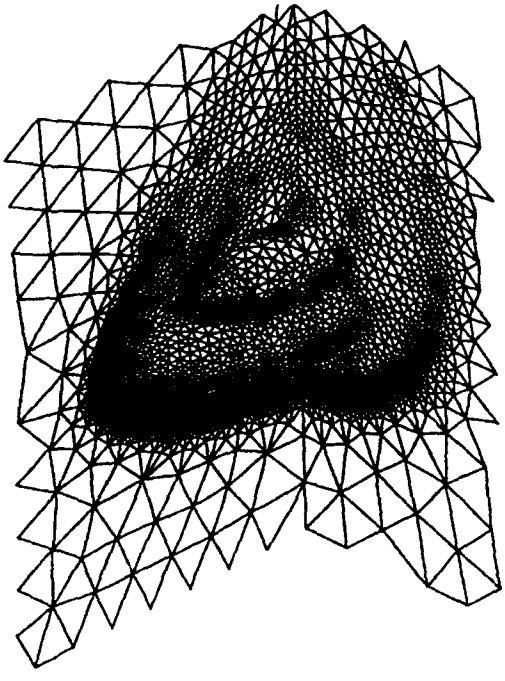


Figure 4.6.3 Comparison of Mach numbers for shock-shock interactions with central and upwind discretizations.



First mesh
119,440 elements
22,032 points





Adapted mesh
275,536 elements
49,715 points

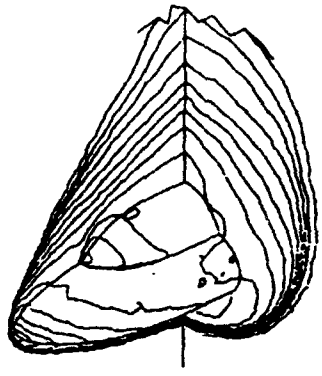


Figure 4.6.4 Initial and adapted unstructured mesh and pressure contours for the hypersonic flow over a double ellipsoid; $M_\infty = 8.15$, $\alpha = 30$ deg.

Isotherms (left, T_∞ = 205.3K) and Iso-Mole Fraction of O_2 (right) Symmetry plane
Euler solutions, M_∞ = 25, α = 30°, double ellipsoid, reactive equilibrium flow

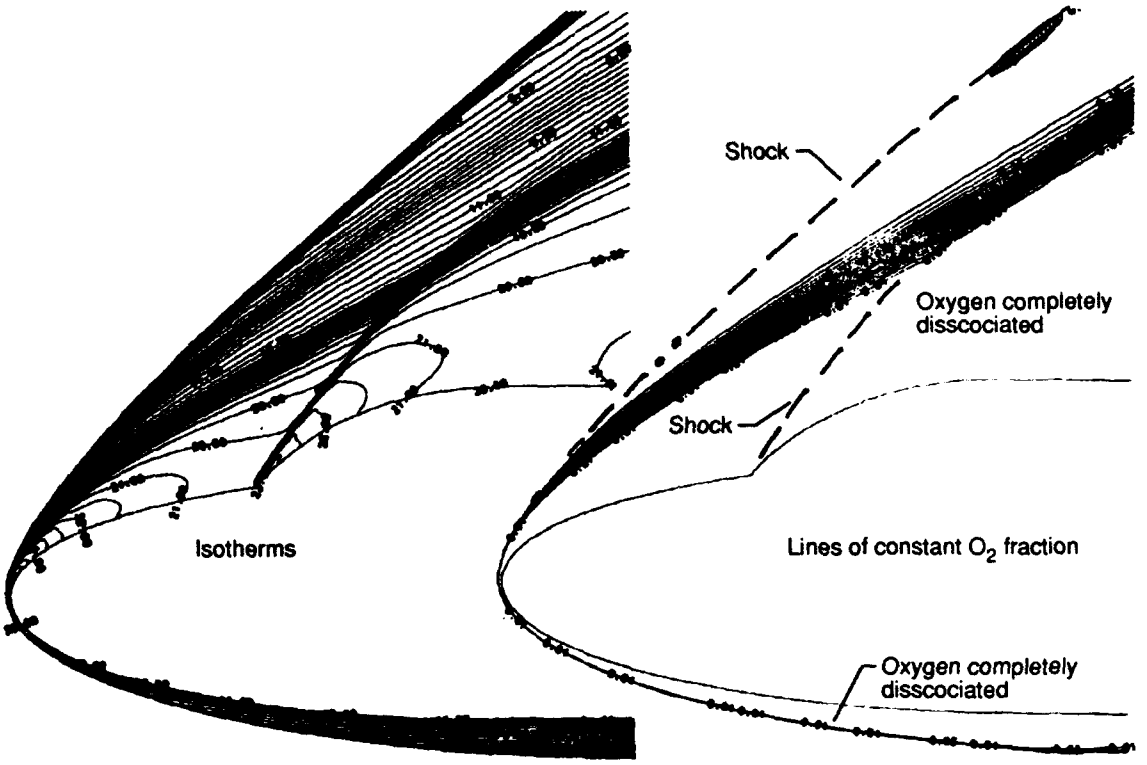


Figure 4.6.5 Real gas effects on a double ellipsoid at hypersonic speeds; M_∞ = 25, α = 30 deg.

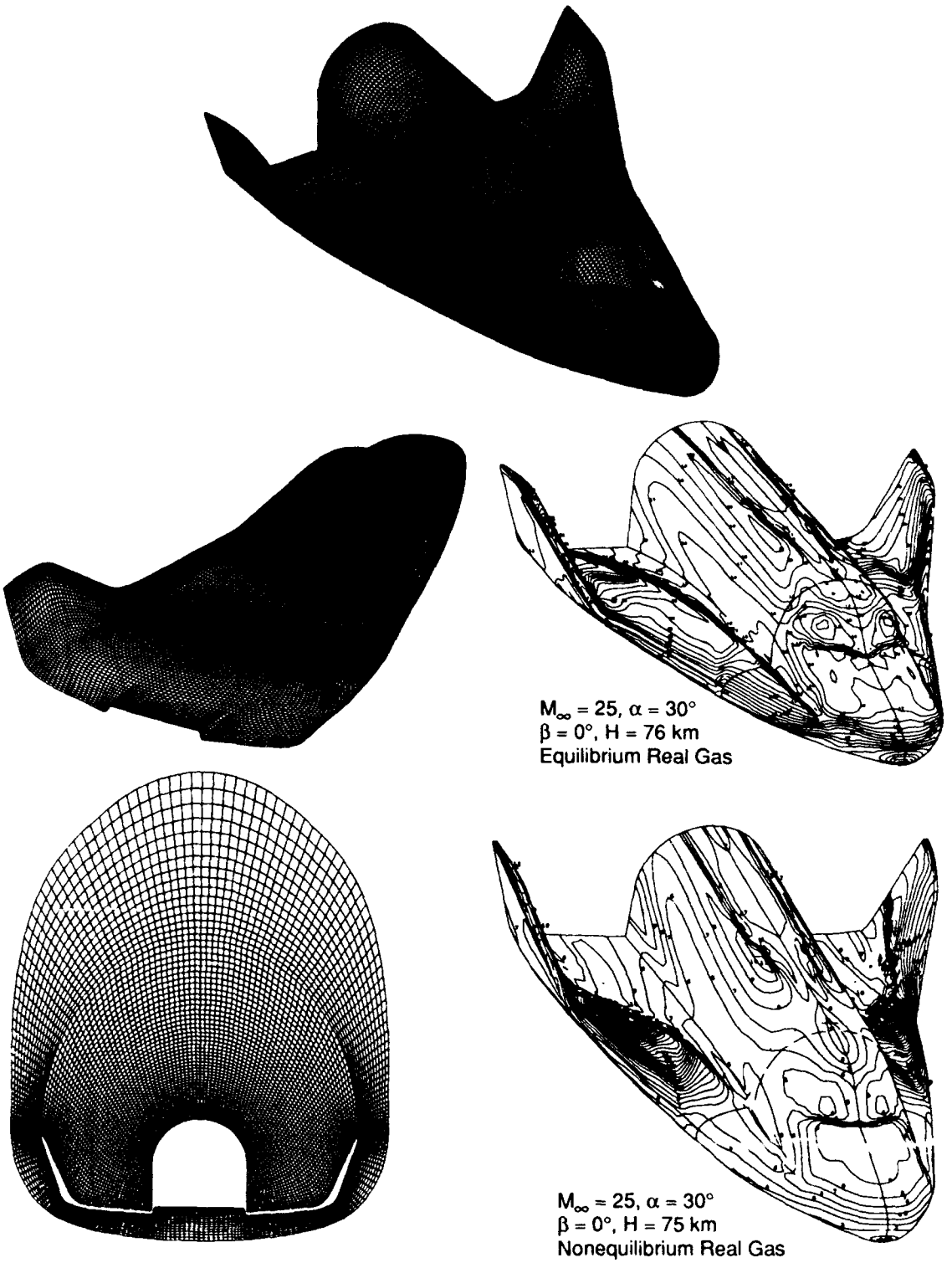


Figure 4.6.6 Euler computations for HERMES 1.0 configuration.

4.7 UNSTEADY FLOWS

A general review of the current status of computational methods for unsteady aerodynamics and aeroelasticity is given by Edwards and Malone in Ref. 1. The demands imposed on the computation of aeroelastic applications such as flutter boundary predictions are intensive, so much so that linear and potential methods are used almost exclusively in the current aircraft design cycle. For example, Edwards and Malone[1] show for transonic, low-α dynamic computations for a wing that a nearly two-orders-of-magnitude computer time penalty is expected when advancing from the transonic small-disturbance (TSD) potential equations to the Euler equations; Navier-Stokes solutions require yet another order-of-magnitude increase. Euler and Navier-Stokes methods are being steadily improved into tools usable to the aeroelastic designer, through the incorporation of more efficient and stable time-integration algorithms and the advent of faster computers. Validation computations to assess the viability of using the Euler equations as opposed to the potential equations have been made; examples of forced oscillation computations for the F-5 wing made with two different Euler solvers are shown below in Figs. 4.7.1–4.7.2. A recent three-dimensional flutter calculation using an unstructured-grid Euler method is also shown. The use of spatial and temporal adaptive-grid schemes are expected to have a significant payoff in this area, and one example is shown below for two-dimensional flow.

4.7.1 Forced Oscillations

F-5 Wing

Unsteady calculations have been performed for forced sinusoidal pitching motion for the F-5 wing (sketched in Fig. 4.7.2) pitching harmonically about a line perpendicular to the root midchord. The pitching motion is described by $\alpha = \alpha_0 + \alpha_1 \cos(\omega t)$, whereas α_0 represents the mean value of angle of attack and α_1 corresponds to the amplitude. The reduced frequency k is defined as $k = \omega c/(2U_\infty)$, where c is based on the mean aerodynamic chord length. The Euler computations of Ref. IV.2.8 correspond to $\alpha_0 = 0.$ deg, $\alpha_1 = 0.109$ deg and $k = 0.274$. Fig. 4.7.1 shows the real and imaginary components of the unsteady pressure distributions at the same three span stations as the steady results shown previously for Euler and modified TSD solvers. On the upper surface, there is a shock pulse in the calculated pressure distributions near 50-60 percent chord, which is produced by the motion of the shock wave. The experimental data (Ref. IV.2.9) does not show a shock pulse in the pressures at the two inboard stations, commensurate with the absence of the shock at these stations in the steady computations. On the lower surface, there are positive and negative spikes in the real and imaginary pressure distributions, respectively, which are much more pronounced in the outboard region of the wing. These spikes are produced by an embedded region of supersonic flow. In general, the two sets of calculated pressures agree well, except near the upper surface shock pulse and in the midchord region along the lower surface. These differences may be attributed to the sharper shock-capturing ability of the Euler code. Also, comparisons with the experimental data are qualitatively good for both the Euler and potential results.

Generally favorable agreement between the Euler and TSD calculations incorporating both entropy and vorticity corrections have been found. For TSD calculations, the grid remains fixed for calculating both steady and unsteady flows so that computations over complex configurations are relatively straightforward. Similar calculations with the Euler equations are complicated by the need for new grids at each time step. Also, since the solution of the three dimensional Euler equations involves five unknowns at each grid point, the computer time required for the Euler calculations is much higher than that required by the solution of the TSD equation. For the Euler code used in the present study, the computational rate for three-dimensional calculations was approximately 60 microseconds/grid point/iteration; the TSD code required only 5 microseconds/grid point/iteration. Both the Euler and TSD calculations were done on a CRAY-2 supercomputer at the National Aerodynamic Simulator facility located at NASA Ames Research Center.

The INFLEX code[2,3] was also applied to the problem of the harmonically oscillating F-5 wing in pitching motion and compared to corresponding measurements. The EUFLEX method was extended as a time-accurate inviscid simulation method, called INFLEX, by Brenneis and Eberle.[2] The method is characterized by corresponding Godunov-type flux formulations known from the basic Eberle Euler method and a first-order accurate backward Euler-type time discretization. Time integration is performed by solving the unfactored implicit operator by a point Gauss-Seidel relaxation method using consistent Jacobi flux matrices. The baseline implicit formulation is secured against singularity effects during inversion of the corresponding block-diagonal matrix by a suitable matrix-conditioning procedure combined with local transformation of conservative to nonconservative variables.

The F-5 wing planform, the surface mesh and the airfoil geometry (modified NACA 65-A-004.8 section) is outlined in Fig. 4.7.2. Because the experimental data revealed large aeroelastic deformations during the pitching cycle, the assumption of a rigid body motion was dropped, and the measured aeroelastic mode shape (Ref. IV.2.9) was used to simulate the real body motion.

The calculation of the transonic test case ($M = 0.95$, $k = 0.132$, $\alpha_0 = 0.0$ deg, $\alpha_1 = 0.523$ deg) was performed on a grid with 106x54x58 points and a CFL number equivalent to 100 Δt/cycle. The comparison between the computed mean surface pressure distribution and the experimental one is favorable at all span stations (Fig. 4.7.2). The suction peak at the leading edge on the lower surface due to the droop nose is reproduced exactly. Both shocks on lower and upper surfaces are overpredicted and shifted aft of the experimental positions, as expected from inviscid simulations. Consequently, the peaks in the real and imaginary parts of the pressures are overpredicted and too far downstream. Except for these small deviations, the results from computation and measurement correlate very well.

LANN Wing

An application of the INFLEX code to a typical transport-aircraft type supercritical wing with an aspect ratio of 7.92 (known as the LANN wing[4,5]) is found in Ref. 6. The wing was experimentally investigated at the NLR facilities and performs harmonic rigid solid body pitching oscillations about the axis normal to the wing root section. The oscillation

is characterized by the mean angle of attack α_0 = 0.6 deg, the amplitude α_1 = 0.25 deg, and a reduced frequency k = 0.132.

Details of the planform with the location of the six spanwise sections for the pressure evaluations and details of the wing tip surface mesh are shown in Fig. 4.7.3. The supercritical root profile section and the wing tip section are also displayed. The pitching axis is located at 62.1 percent of the root chord from the wing apex. The position of the wing inside the computational domain is sketched; the calculations were performed on a H-type mesh with 80x38x46 grid points, where on the wing 18 grid lines are place in the spanwise direction and 50 grid points are used to discretize the lower and upper side of the profile section.

The mean pressure distributions of individual wing sections indicate that except for a slight overprediction on the upper surface behind the shock, the overall agreement of numerical results to experimental data is quite good. The computation of further tests cases confirm this statement. However, at higher Mach numbers, shock boundary-layer interaction effects are responsible for systematic deviation of the shock positions from experimental findings.

Spatial and Temporal Adaption

In order to reduce the computational time associated with aeroelastic solutions, adaptive grid methods that are adaptive in both space and time can be used.[7,8] Two-dimensional calculations are shown in Fig. 4.7.4 for one such adaptive method applied to a pitching NACA 0012 airfoil. The baseline Euler method is an upwind, explicit finite-volume scheme. The computational grid and the density contours at several points in the cycle of oscillation are shown. The mesh adapts spatially and temporally to the aerodynamic response to the oscillation; the comparison with experiments and other established schemes for this case indicates that highly accurate solutions can be obtained with a significant savings in computer time over standard global time-stepping schemes.

4.7.2 Flutter Predictions

The flutter predictions from a time-marching aeroelastic procedure,[9] which couples an implicit, three-dimensional, upwind, unstructured-grid Euler code to the structural equations of motion, is shown in Fig. 4.7.5. The unstructured-grid for the 45-deg sweptback wing was developed using an advancing-front method. The mesh deforms during the calculation due to aerodynamic loading and is modeled as a spring network where each edge of the tetrahedra represents a spring with a stiffness proportional to the edge length. As the surface mesh moves, static equilibrium equations are solved to determine the interior grid points. The implicit scheme is a Gauss-Seidel scheme in which the relaxation is implemented by ordering the elements in a downstream-to-upstream pattern. Large time steps selected on the basis of temporal accuracy of the simulation are possible.

The wing is an AGARD standard aeroelastic configuration which was tested in the Transonic Dynamic Tunnel[10] at NASA Langley Research Center. The wing is modeled structurally using the first four natural vibration modes. The experimental flutter speed index and the nondimensional flutter frequency as a function of free-stream Mach number define a typical transonic flutter dip with the bottom of the dip near sonic conditions. The computed results agree well with the experimental data at M_∞ = 0.499 and 0.678 in flutter speed index and in frequency. Near the transonic flutter dip, the computations differ from experiment in flutter speed index, but agree reasonably well in flutter frequency ratio. Robinson et al.[11] present Euler computations using a structured-grid code for this case, which agrees closely with that presented. Other results for a supersonic transport configuration are given elsewhere.[9]

4.7.3 References

1. EDWARDS, J. W. and MALONE, J. B., "Current Status of Computational Methods for Transonic Unsteady Aerodynamics and Aeroelastic Applications," Computing Systems in Engineering, Vol. 3, No. 4, 1992, pp. 545–569.

2. BRENNEIS A., and EBERLE, A., "Application of an Implicit Relaxation Method for Solving the Euler Equations for Time-Accurate Unsteady Problems," Journal of Fluids Engineering, Vol. 112, 1990, pp. 510–520.

3. BRENNEIS, A., "Berechnung Instationarer Zwei- und Dreidimensionaler Stromungen um Tragfluegel Mittels eines Impliziten Relaxationsverfahrens zur Losung der Eulergleichungen," FORTSCHRITT-BERICHT VDI 7/165, Dusseldorf, 1989.

4. ZWANN, R. J., "LANN-wing. Pitching Oscillation. Data Set 9," AGARD-R-702, 1982.

5. HORSTEN, J. J., DEN BOER, R. G. and ZWANN, R. J., "Unsteady Transonic Pressure Measurements on a Semi-Span Wind Tunnel Model of a Transport-Type Supercritical Wing (LANN model)," AFWAL-TR-83–3039, Part I and II, 1984.

6. SCHMATZ, M. A., BRENNEIS A., and EBERLE, A., "Verification of an Implicit Relaxation Method for Steady and Unsteady Viscous and Inviscid Flow Problems," Paper 15 in AGARD CP–437, Vol. 1, 1988.

7. HOOKER, J. R., BATINA, J. T. and WILLIAMS, M. H., "Spatial and Temporal Adaptive Procedures for the Unsteady Aerodynamic Analysis of Airfoils Using Unstructured Meshes," AIAA 92-2694, June 1992.

8. RAUSCH, R. D., BATINA, J. T. and YANG, H. T., "Spatial Adaptation Procedures on Tetrahedral Meshes for Unsteady Aerodynamic Flow Calculations," AIAA 93-0670, Jan. 1993.

9. RAUSCH, R. D., BATINA, J. T. and YANG, H. T. Y., "Three-Dimensional Time-Marching Aeroelastic Analyses Using an Unstructured-Grid Euler Method," AIAA Journal, Vol. 31, No. 9, 1993, pp. 1626–1633.

10. YATES, E. C., Jr., LAND, N. S. and FOUGHNER, J. T., Jr., "Measured and Calculated Subsonic and Transonic Flutter Characteristics of a 45° Sweptback Wing Planform in Air and in Freon-12 in the Langley Transonic Dyanamics Tunnel," NASA TN D-1616, March 1963.

11. ROBINSON, B. A., BATINA, J. and YANG, H. T. Y., "Aeroelastic Analysis of Wings Using the Euler Equations with a Dynamic Mesh," AIAA 90–1032, April 1990.

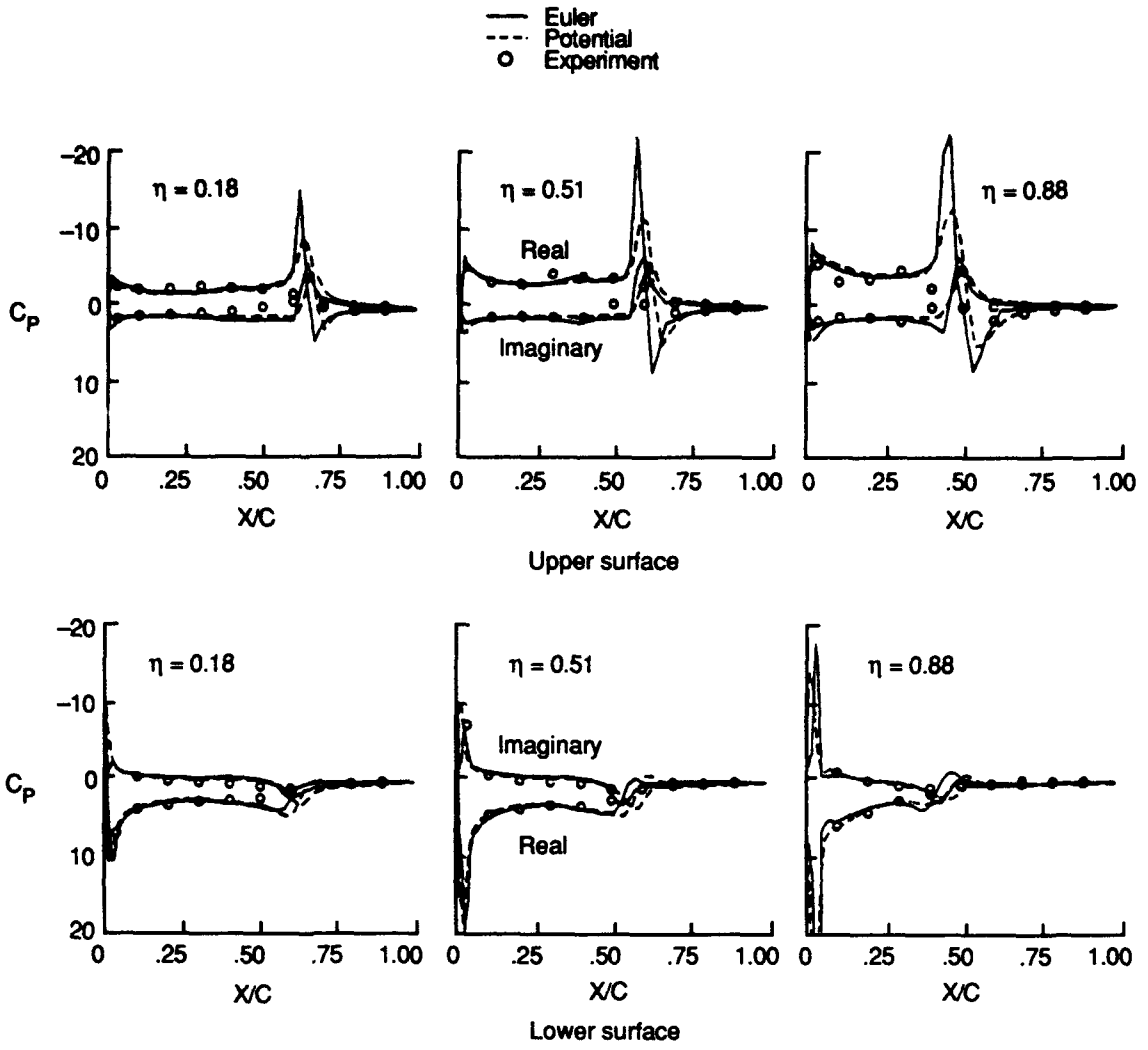
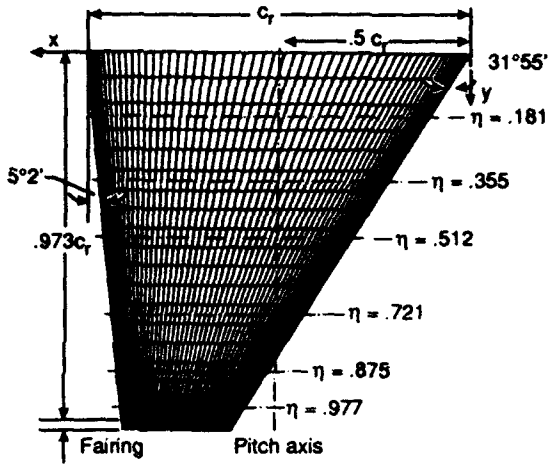
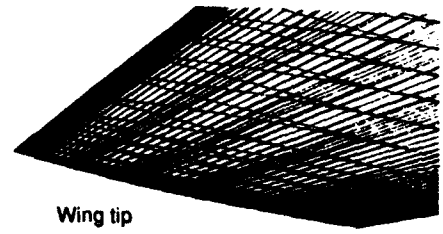


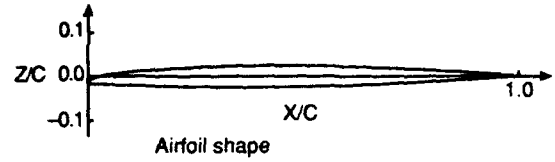
Figure 4.7.1 Comparison of first harmonic components of the pressure distribution for the F-5 wing; M_∞ = 0.9, α_0 = 0.0 deg, α_1 = 0.109 deg, k = 0.274.



Planform and surface mesh

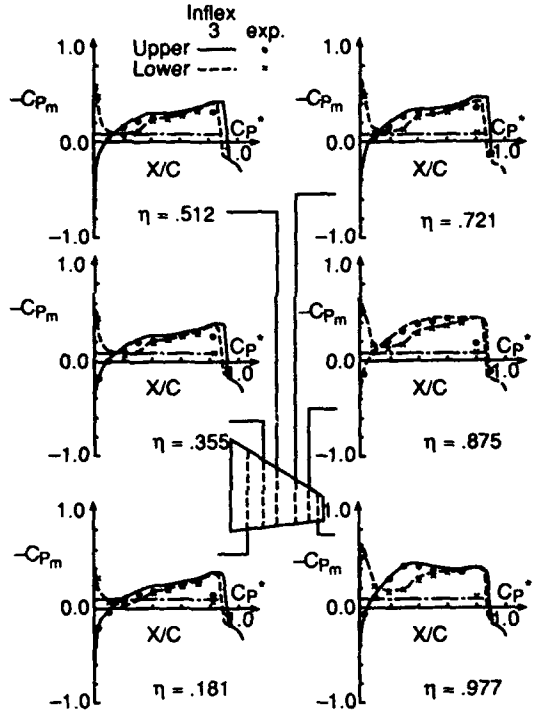


Wing tip



Airfoil shape

Mean pressure distributions



In-phase component pressure distributions

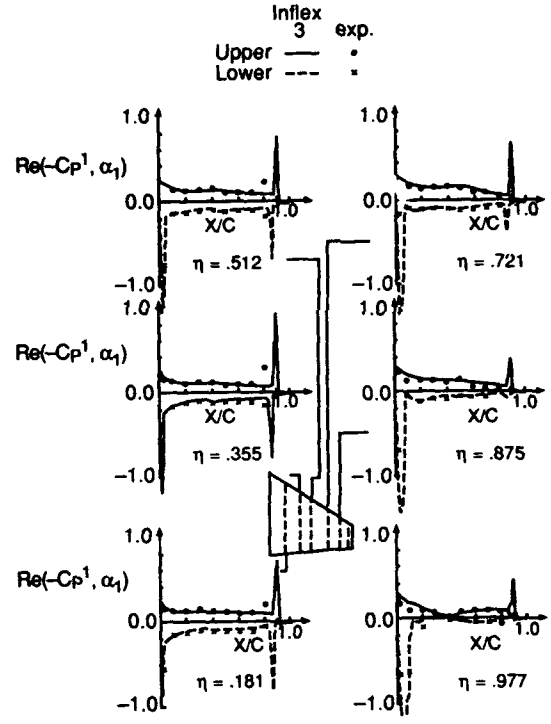


Figure 4.7.2 Spanwise comparison of the measured and calculated mean and in-phase component pressure distributions for the F-5 fighter wing; M_∞ = 0.95, k_r = 0.132, α_0 = 0.0 deg, α_1 = 0.523 deg.

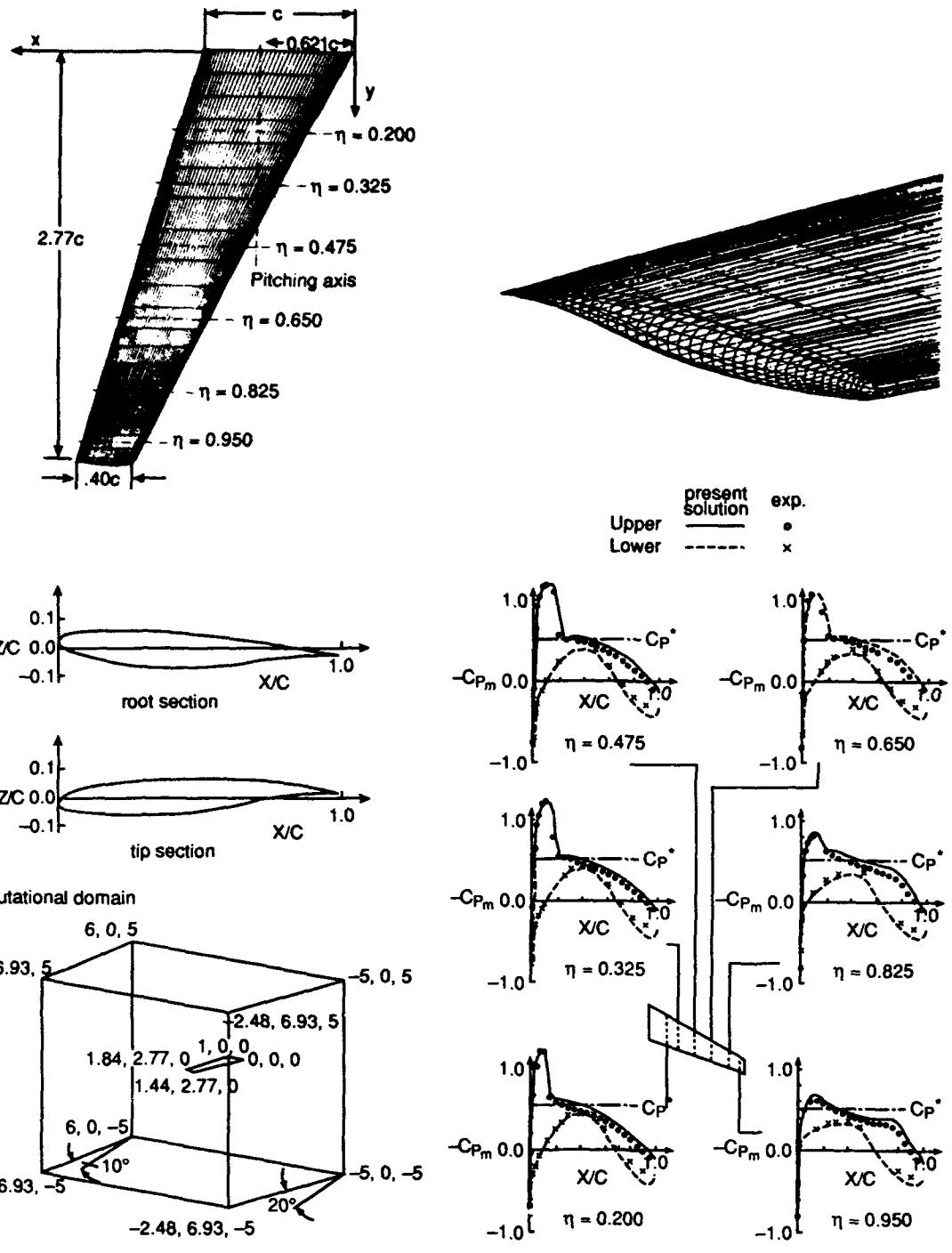


Figure 4.7.3 Spanwise comparison of measured and calculated mean pressure distributions for the LANN-wing; M_∞ = 0.77, α_0 = 0.6 deg, α_1 = 0.25 deg, k = 0.132.

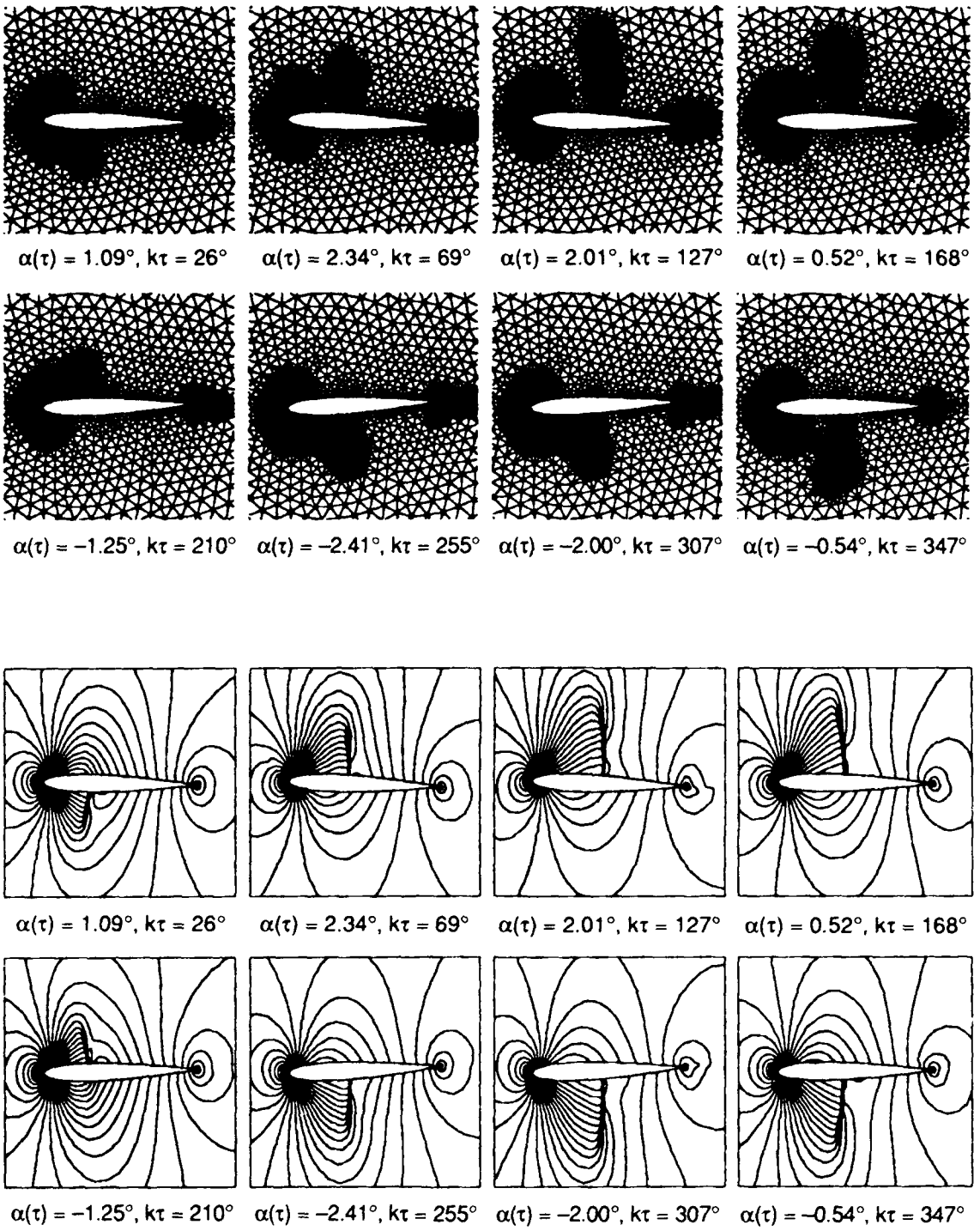
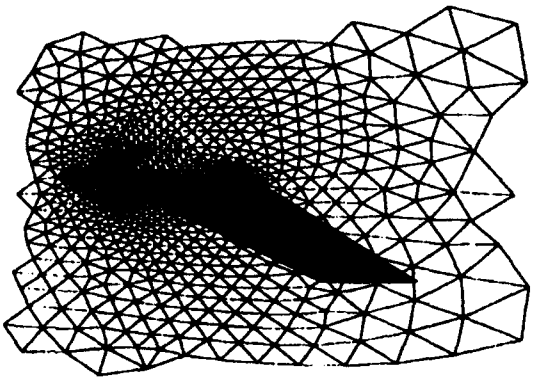
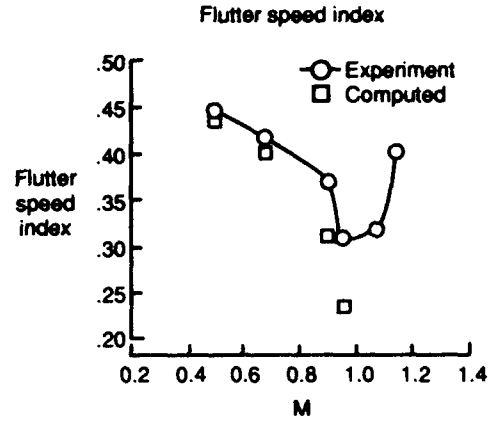


Figure 4.7.4 Instantaneous density contour lines obtained from the spatial and temporal adaption procedure for the NACA 0012 airfoil pitching harmonically at M_∞ = 0.755, α_0 = 0.016 deg, α_1 = 2.51 deg, k = 0.0814







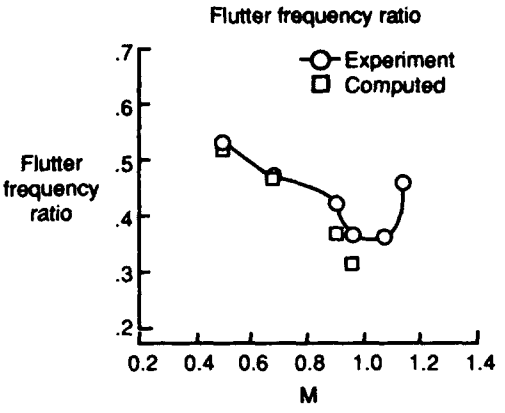


Figure 4.7.5 Partial view of the surface mesh and the symmetry plane mesh and comparisons of Euler flutter predictions with experimental data for a 45 deg swept-back wing.

4.8 DESIGN APPLICATIONS

Examples of design applications using the Euler equations are shown below. A summary of aerodynamic design methods is given by Sloof[1] and Dulikravich.[2] Much of the design process of aircraft is conducted using repeated analyses of candidate geometries constructed through the background knowledge and experience of the designer. In such an approach, the turnaround for analysis must be rapid and, until recently, the design of aircraft components at transonic speeds was done almost exclusively using potential methods coupled with boundary-layer schemes. Euler methods were used to assess component interference problems and to address flows at higher Mach numbers, where the potential equations are inadequate. With faster turnaround of the Euler and Navier-Stokes solvers, these methods are being used more frequently.[3] The parametric approach is shown below for the design of a porous airfoil at transonic speeds. Of automated approaches, there are two general methods that are being used currently: (1) global optimization methods in which a number of trial shapes are examined to arrive at a minimum of an objective function and (2) inverse design procedures in which the local geometry is changed in order to match a prescribed variation of the pressures. The optimization approach is shown for the design of conical wings and low-drag bodies; the inverse approach is shown for the design of a supercritical airfoil and an engine nacelle.

4.8.1 Airfoils

Supercritical Airfoil

Drela[4] developed a two-dimensional mixed inverse procedure where the pressure is prescribed over a portion of the geometry, *denoted as a freewall segment*, and the geometry prescribed on the other. The procedure is an extension of an analysis method which uses a direct solver for the Euler equations coupled to an integral boundary-layer method. An example of this capability is shown in Fig. 4.8.1, in which the wave drag is reduced for a transonic airfoil by weakening the shock wave in the flowfield. A mixed inverse calculation was performed for the RAE 2822 Case 6 airfoil[5] with the freewall segment encompassing nearly the entire suction surface. Starting with the analysis pressure distribution, a smoothed-out pressure distribution was somewhat arbitrarily specified over the freewall segment.

The mixed inverse calculation was started from the analysis solution, and required five additional Newton iterations to converge to machine zero. The boundary layer coupling option was retained for this calculation. The resulting pressure distribution from the inverse output differs slightly from the inverse input because the geometry is constrained to be continuous at the freewall segment endpoints. Fig. 4.8.1 shows the geometry comparison between the original and modified airfoils and the Mach number contours for the original and modified solutions. The shock has indeed been eliminated, resulting in a substantial drag reduction.

Porous Transonic Airfoil

In the early 1980's, a number of experimental and computational studies were conducted to look at the advantages of delaying the drag rise of airfoils by venting the shock through a porous surface,[6] as shown schematically in Fig. 4.8.2. The porous surface induces a separation bubble in the vicinity of the upper surface shock and an oblique compression wave forms which decreases the strength of the terminating normal shock. The drag rise is reduced when the resulting energy loss associated with the normal shock and the separated boundary layer is less than that for the nonporous airfoil.

Hartwich[7] studied computationally the effect of shock venting for the NACA 0012 and a supercritical-type section. The surface boundary conditions were modeled to induce a normal velocity determined by the difference between the surface pressure and the cavity pressure. In general, this velocity would be determined by viscosity and the porosity of the surface, but the approximate boundary condition and the Euler equations were used in lieu of viscous solutions because of the uncertainty associated with current turbulence models. The porosity level σ was varied for both airfoil sections. A typical pressure distribution for the porous and nonporous NACA 0012 section is shown. The lift is increased, in some instances by 65 percent. The lift and wave drag indicate that an order of magnitude reduction in wave drag at constant lift has been attained for overspeed conditions (low lift levels). The supercritical section results also demonstrated a reduced drag at constant lift. The extents of porous areas used are greater than those considered previously; the results demonstrate a potential for designing a transonic airfoil for reduced drag at multiple design points.

4.8.2 Engine Nacelles

A fast, efficient and user-friendly inverse design system for three-dimensional nacelles is in use at General Electric.[8] The system allows the fan cowl designer to modify either all or a portion of the three-dimensional fan cowl. The designer specifies the target pressure distribution on the crown, side, and keel cuts of the nacelle, as in Fig. 4.8.3. A modification of the predictor/corrector design approach[9] enables the geometry to be altered based on the difference between the calculated and target pressures. A number of example applications for the design of both axisymmetric and three-dimensional nacelles is given in Ref. 8.

The design method[9] uses two design algorithms, one for subsonic flow and the other for supersonic flow. The supersonic algorithm is blended with the subsonic algorithm to design regions of transonic flow. Both algorithms assume that ΔC_p is proportional to the change in geometry. The subsonic algorithm is based on the assumption that changes in curvature are directly proportional to changes in pressure coefficient. The relationship used to express the change in curvature as a function of change in pressure coefficient is:

$$\Delta C = \Delta C_p A (1 + C^2)^B$$

where

C is the curvature

C_p is the pressure coefficient

A = +1/−1 for the upper/lower surface

B = input constant ranging from 0.0 to 0.5

The change in curvature is converted to a change in r'' through a relation valid for small changes in the surface slope.

$$\Delta r'' = \Delta C_A \left(1 + r'^2\right)^{1.5}$$

where r, r' and r'' are the surface radius, slope, and curvature, respectively.

The supersonic algorithm is based on supersonic thin airfoil theory. Based on relations between the pressure coefficients and surface slope the expression

$$\Delta r' = K \Delta C_p$$

can be derived.[9] Differentiating this expression gives the following relationship between r'' and ΔC_p.

$$\Delta r'' = K \frac{d(\Delta C_p)}{dx}$$

The value for the constant K is 0.05 and is used to under relax the changes in the geometry during each design iteration.

The required change in curvature is thus calculated at each point along the fan cowl. To ensure geometrical closure at the downstream station, the procedure of Lin et al.[10] is used. A sine function is added to the target pressure with the maximum modification at the center of the design region and zero at the ends. The amplitude of the sine function is iteratively determined to close the geometry by modifying the target pressure distribution.

The results for the three-dimensional test case shown in Fig. 4.8.3 correspond to a design range starting at the nacelle leading edge and ending 10 inches upstream of the nacelle trailing edge. The Mach number distributions along the crown, side and keel cuts of the original nacelle as well as the desired target Mach number distribution are shown. The target Mach number distribution was achieved after 40 design iterations; the resulting modified geometry is shown on an enlarged vertical scale.

4.8.3 Supersonic Conical Wings

A procedure for the design of wings at supersonic speed based on a numerical optimization technique[11] coupled with a solution scheme for the Euler equations is given in Ref. 12. The wings considered can be either conical or three-dimensional delta wings with a straight leading edge. The surface is given by a set of Legendre polynomials. The coefficients of this set are the design parameters in the optimization task; the object function is the lift-to-drag ratio. Results of the optimization are shown for conical wings at an onflow Mach number of M_∞ = 4.8. The convergence of the optimization process is monitored as a function of the number of numerical L/D computations and the number of design variables. The influence of geometrical and aerodynamic parameters on the optimization result is examined in Fig. 4.8.4. An example for the design of a three-dimensional wing is given in Ref. 12.

4.8.4 Low-Drag Bodies

The optimization code COPES[12] was combined with an Euler space-marching method by H. Rieger[13] for the analysis of two– and three–dimensional bodies in supersonic flow. The fundamental equations to be solved are the conservation laws in integral form. By restriction to purely supersonic flows the problem becomes hyperbolic. By use of the balance of flux values across the surfaces of finite volumes, all flux values of one finite volume layer normal to the stream direction can be deduced from the values of the preceding layers. This allows the application of a Runge-Kutta integration method to the flux values in downstream direction.

Although the present method can be used for more general three–dimensional bodies,[14] here only bodies of revolution are considered. These are of special interest because Miele[15] presents some optimized shapes derived under special assumptions (linearized potential equation, slender body simplification). To convert the contour optimization into a parameter optimization problem, the radius of the body of revolution is represented by the superposition of a constant and Legendre polynomials. The coefficient of this superposition are take as design variables. Only the constant and the last Legendre coefficient are analytically determined in order to get a pointed nose and a desired base radius.

To demonstrate the efficiency of the combined code, the following optimization problem was examined:

design objective: Find a closed and pointed body of revolution with minimum wave drag coefficients C_D (referred to actual cross section area).

constraint: Least volume $V/c^3 \geq 0.005$.

design variables: The contour is approximated by superposition of the first five Legendre polynomials. The first four coefficients are used as design variables, while the fifth is adapted for the base radius zero.

constant value: M = 3.0.

start-up design: The starting contour is a parabola with the thickness $2\, r_{max}/c = 0.1$ and the volume $V/c^3 = 0.00419$, shown in Fig. 4.8.5.

For this starting design the computational grid was selected fine enough to give reasonable accuracy concerning the wave drag. The corresponding pressure distribution is shown on the left side. The integrated wave drag is $C_D = 0.07289$. In order to save computation time, at the beginning the analysis code was run with a coarser grid and later continued with the fine grid to find the best design. Increasing the body volume, which is too small at the beginning, leads to a growing wave drag.

In Fig. 4.8.5, the resulting optimum body is compared to the Sears-Haack body which is the optimum under the assumption of a linearized potential equation and slender body simplifications. For M = 3.0 there are some differences. For M = 1.5 the above mentioned assumptions are violated less and so the present method optimum is much nearer to the Sears-Haack body.

As the Sears-Haack body has a vertical tangent and therefore a small subsonic flow area at the leading and trailing edge, it cannot be calculated by the present Euler space-marching method. At M = 3.0 some modifications were possible to get the drag value nevertheless. It is considerably higher (3.6 %) than for the best design found even with the restrictions implied in the superposition formula.

4.8.5 References

1. SLOOF, J. W., "A Survey of Computational Methods for Subsonic and Transonic Aerodynamic Design," Proc. International Conference on Inverse Design Concepts in Engineering Sciences, 1984.

2. DULIKRAVICH, G. S., "Aerodynamic Shape Design and Optimization," AIAA 91-0476, Jan. 1991.

3. GARNER, P., MEREDITH, P., and STONER, R., "Areas of Future CFD Development as Illustrated by Transport Aircraft Applications," AIAA 91-1527, June 1991.

4. DRELA, M., "Two-Dimensional Transonic Aerodynamic Design and Analysis Using the Euler Equations," Ph.D. Dissertation, Massachusetts Institute of Technology, Dec. 1985.

5. COOK, P. H., MCDONALD, M. A. and FIRMIN, M. A., "Aerofoil RAE 2822 Pressure Distributions and Boundary Layer and Wake Measurements," AGARD AR-138, 1979.

6. NAGAMATSU, H. T., OROZCO, R. D., and LING, D. C., "Porosity Effect on Supercritical Airfoil Drag Reduction by Shock Wave/Boundary Layer Control," AIAA 84-1682, June 1984.

7. HARTWICH, P. M., "Euler Study on Porous Transonic Airfoils with a View Towards Multipoint Design," AIAA 91-3286, Sept. 1991.

8. UENISHI, K., PERSON, M. S., LEHNIG, T. R. and LEON, R. M., "CFD Based 3D Turbofan Nacelle Design System," AIAA 90-3081, 1991.

9. CAMPBELL, R. L. and SMITH, L. A., "A Hybrid Algorithm for Transonic Airfoil and Wing Design," AIAA 87-2552-CP, Aug. 1987.

10. LIN, W. F., CHEN, A. W. and TINOCO, E. N., "3D Transonic Nacelle and Winglet Design," AIAA 90-3064-CP, 1990.

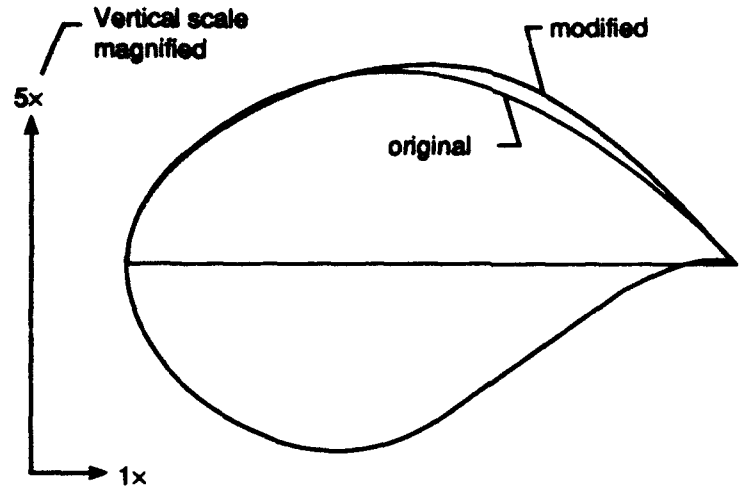
11. SCHONE, J., "Design of Supersonic Wing Using an Optimization Strategy coupled with a Solution Scheme for the Euler-Equations," AIAA-90-3060, 1990.

12. VANDERPLAATS, G. N. "Approximation Concepts for Numerical Airfoil Optimization," NASA TP-1370, 1979.

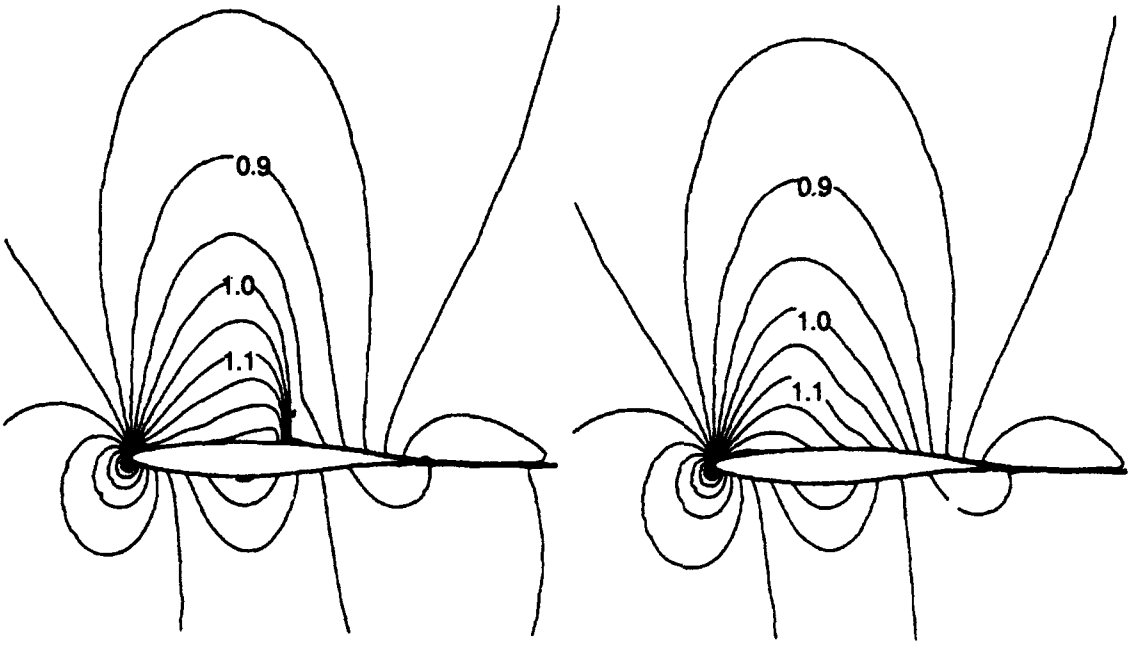
13. RIEGER, H., "Aerodynamics of Hypersonic Lifting Vehicles," AGARD CP-428 (1988).

14. BOCK, K. W., "Aerodynamic Design by Optimization," Paper 20 in AGARD CP-463, 1990.

15. MIELE, A. (editor), "Theory of Optimum Aerodynamic Shapes: External Problems in Aerodynamics of Supersonic Hypersonic and Free-Molecular Flows," Academic Press, New York/London, 1965.

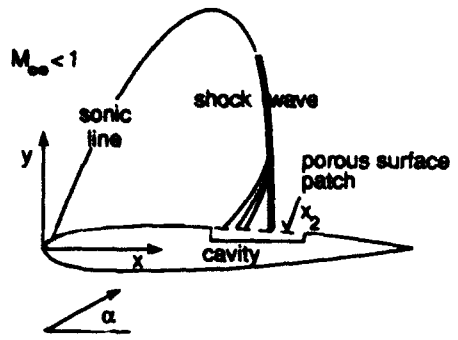


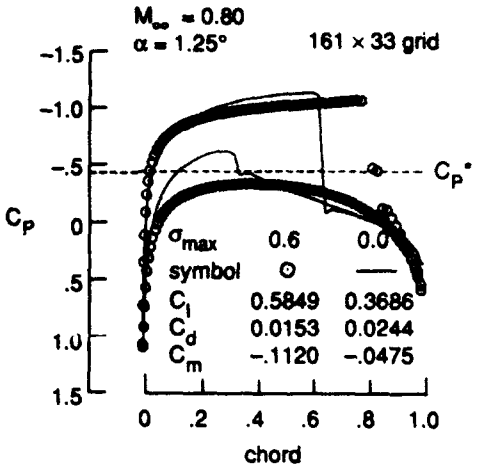
Comparison of RAE 2822 airfoil geometry
before and after mixed redesign

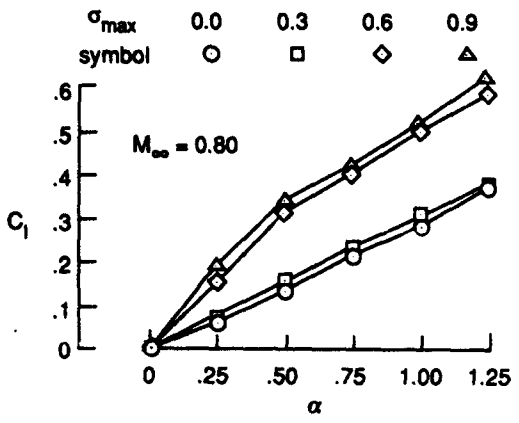


Mach number contours before and after RAE 2822 mixed redesign

Figure 4.8.1 Mixed redesign of supercritical airfoil.







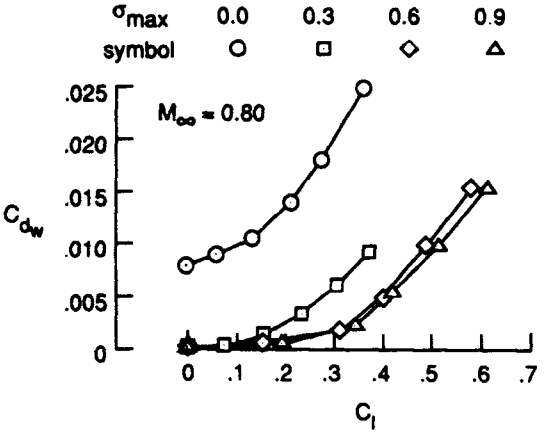
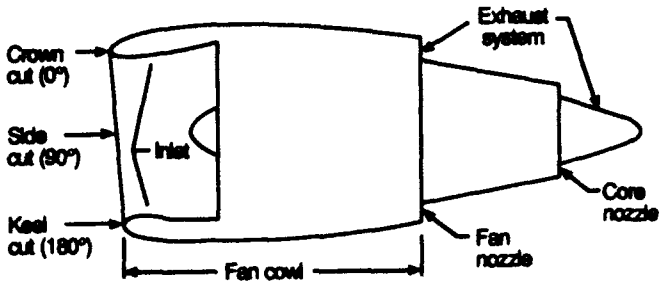
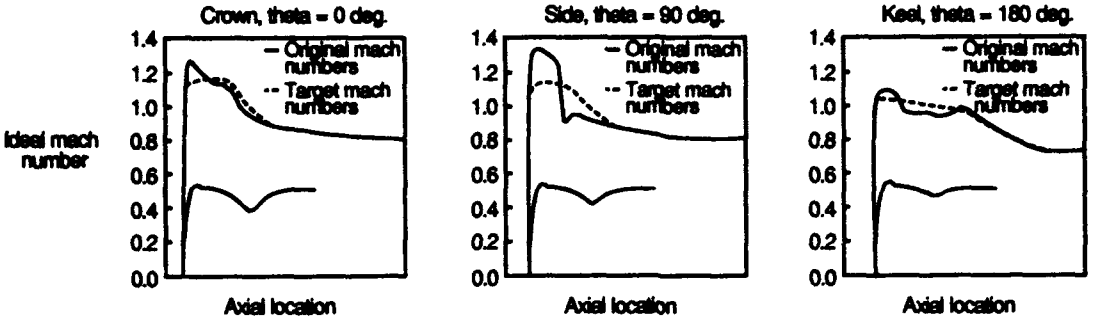
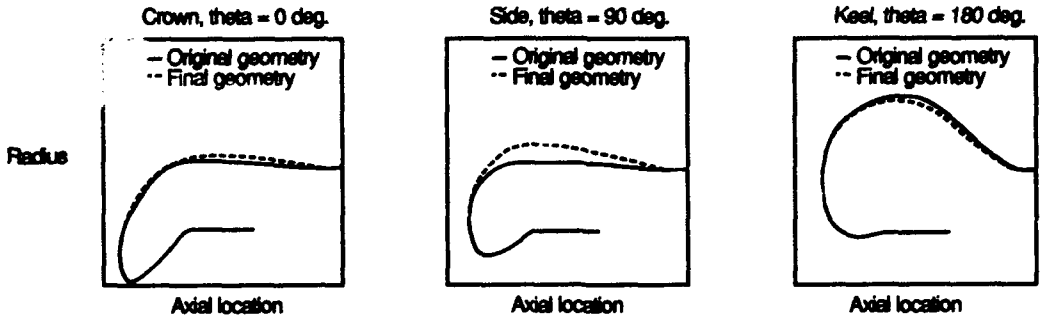


Figure 4.8.2 *Porous airfoil in transonic flow.*



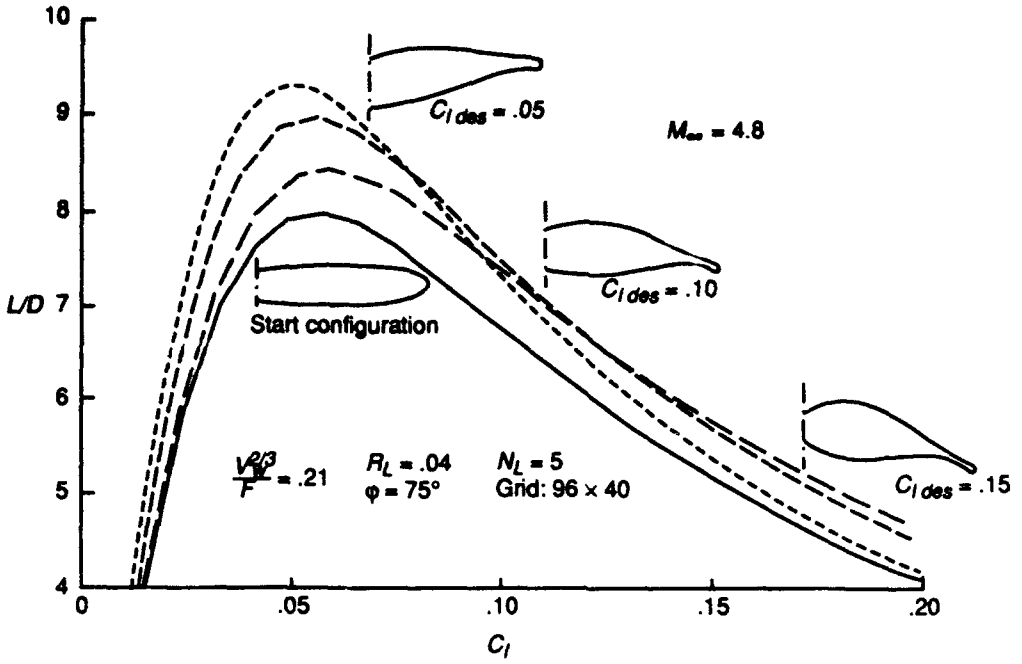


Original and target mach number distributions for the three-dimensional test case



Original and final geometries for the three-dimensional test case

Figure 4.8.3 Nacelle design using the Euler equations.



M = const. ΔM = .1 grid: 96 × 40

R_L = .04 φ = 75° $\frac{V_W^{2/3}}{F}$ = .21 M_∞ = 4.8 C_l = .150

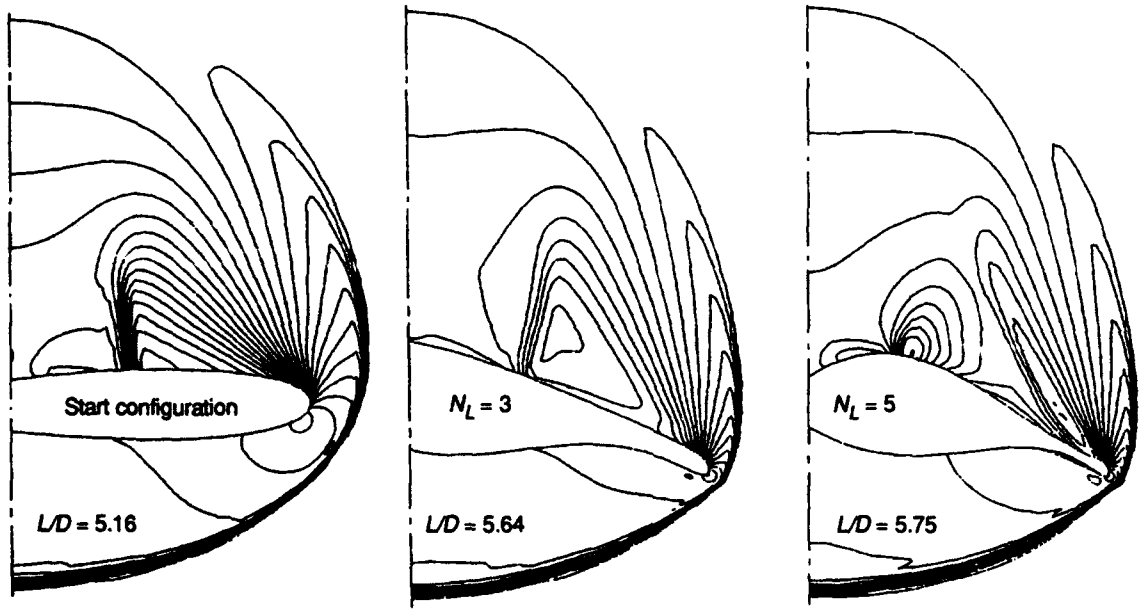
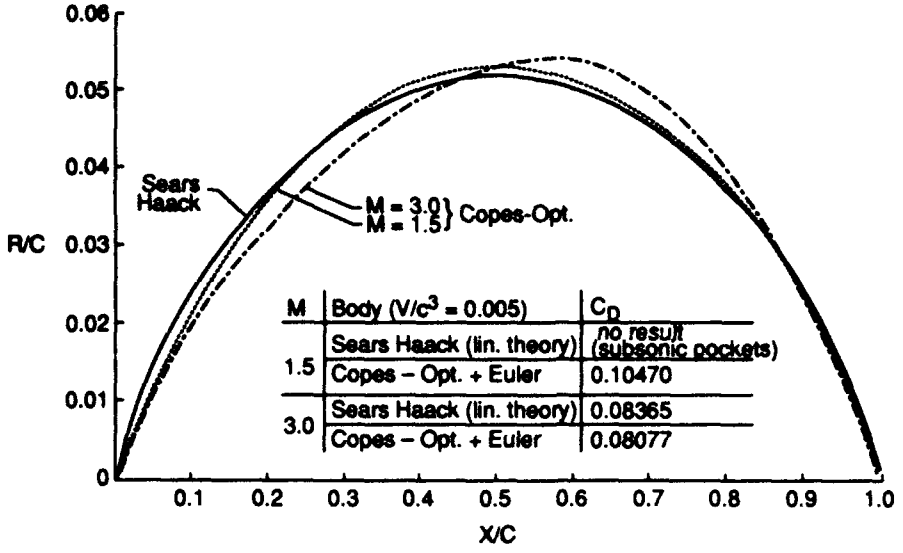
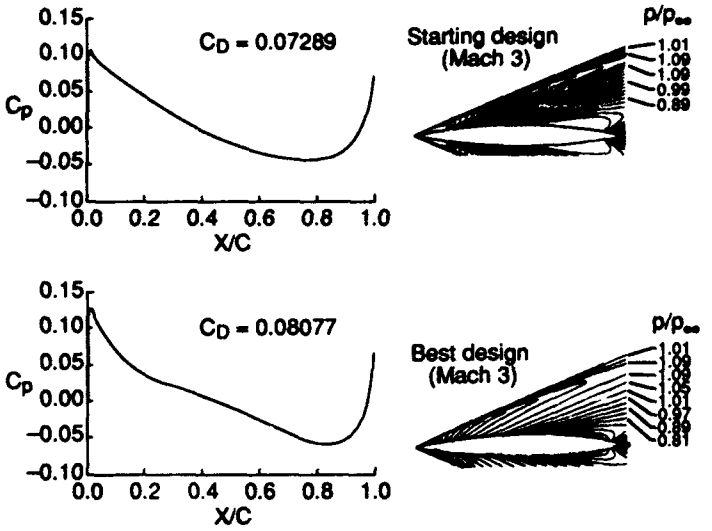


Figure 4.8.4 Optimized conical delta wings.





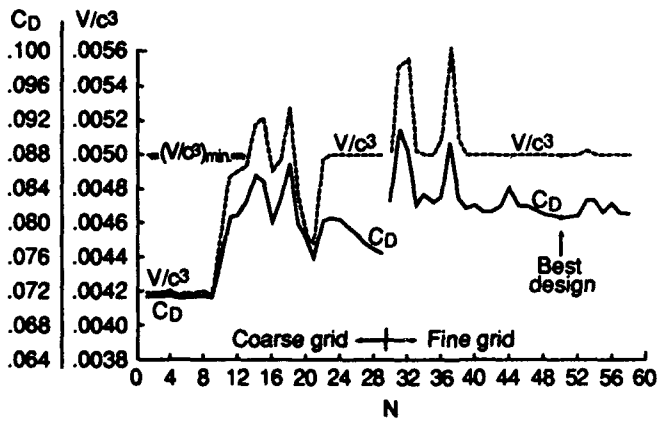


Figure 4.8.5 Body-designs for low drag.

This page has been deliberately left blank.

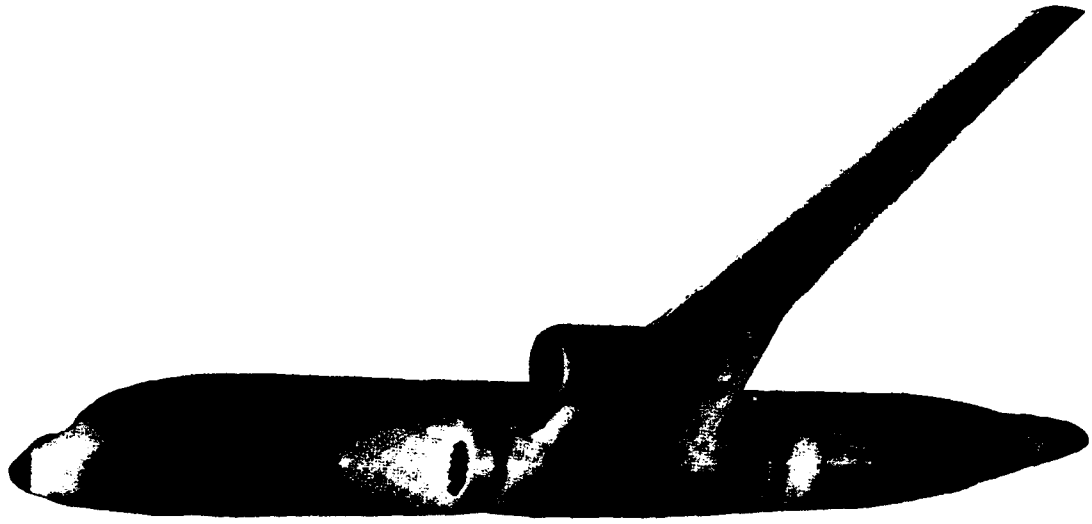


Plate 4.1 Pressure distribution on wing-body-pylon-nacelle configuration with flow-through nacelle.

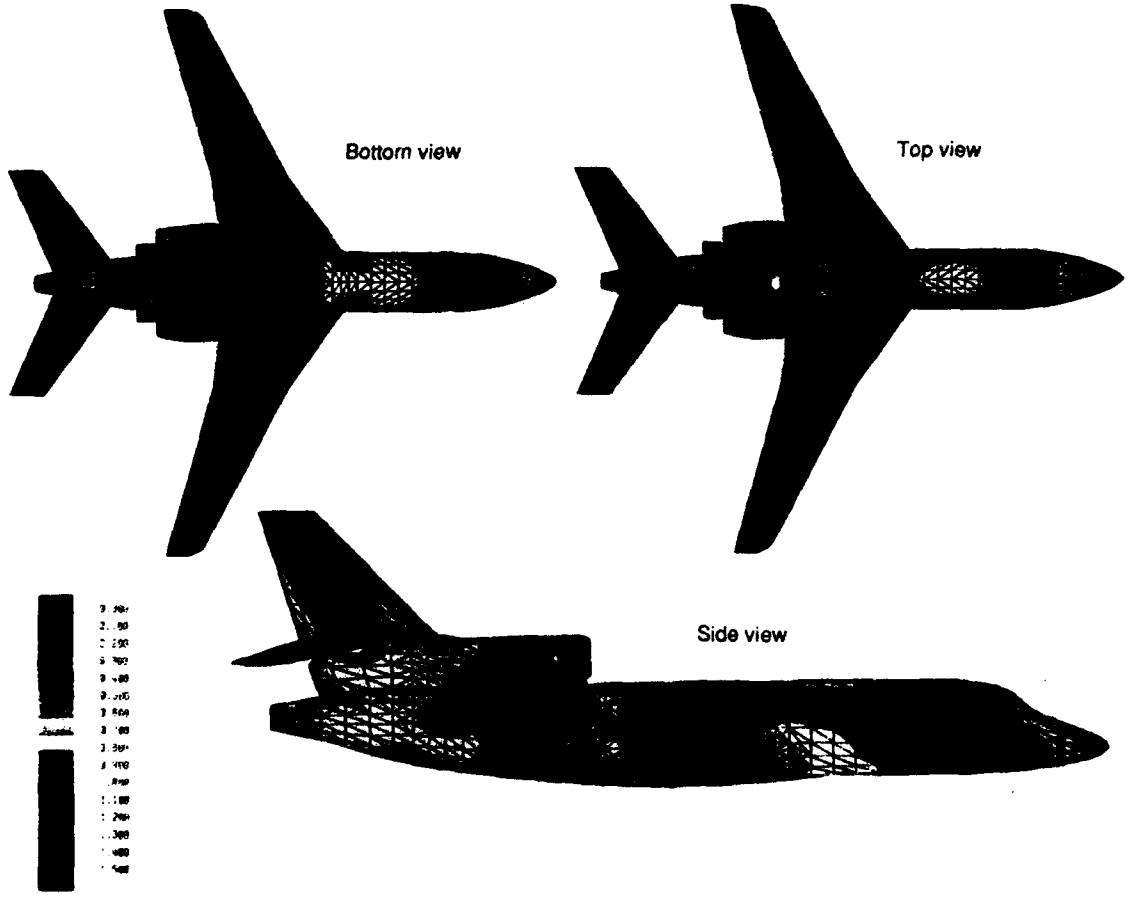


Plate 4.2 Isomachlines of unstructured-grid computation of the Dassault Falcon aircraft; $M_\infty = 0.85$, $\alpha = 3.0$ deg.

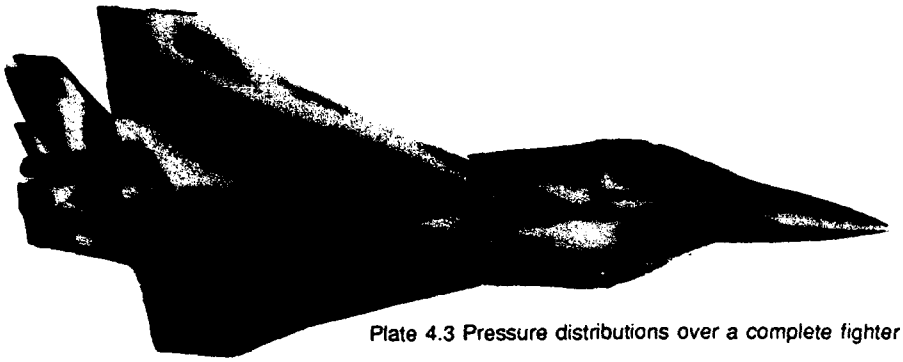


Plate 4.3 Pressure distributions over a complete fighter configuration.

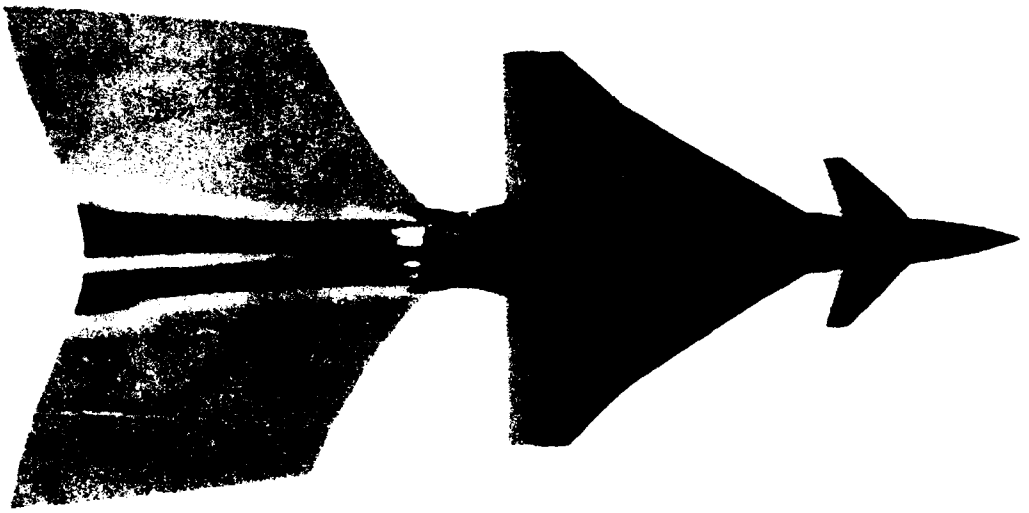


Plate 4.4 Temperature contours over a fighter configuration with propulsion simulation

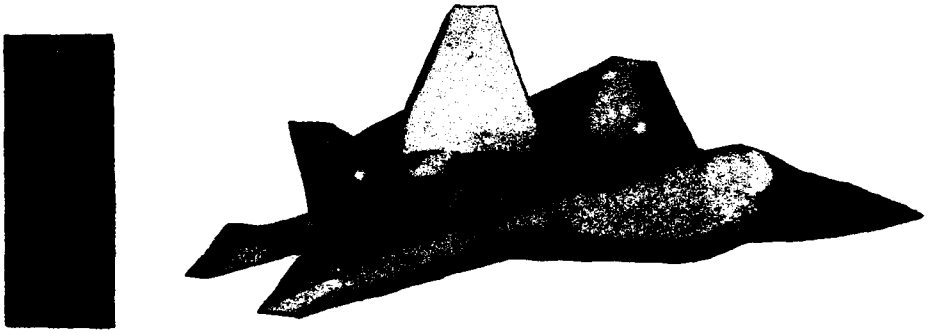
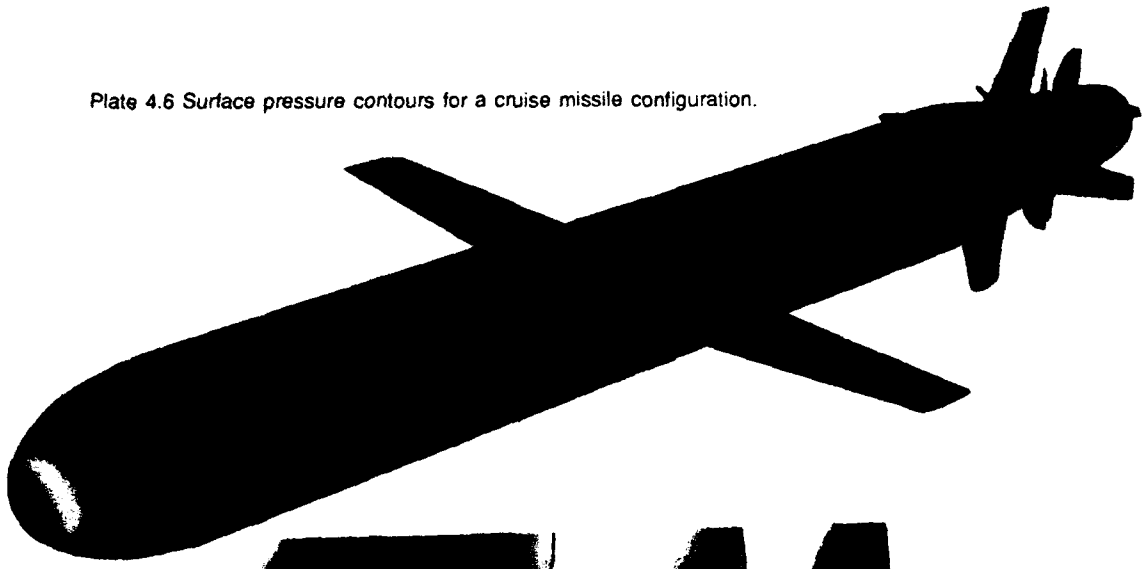


Plate 4.5 Surface pressure contours for an advanced tactical fighter configuration.

Plate 4.6 Surface pressure contours for a cruise missile configuration.



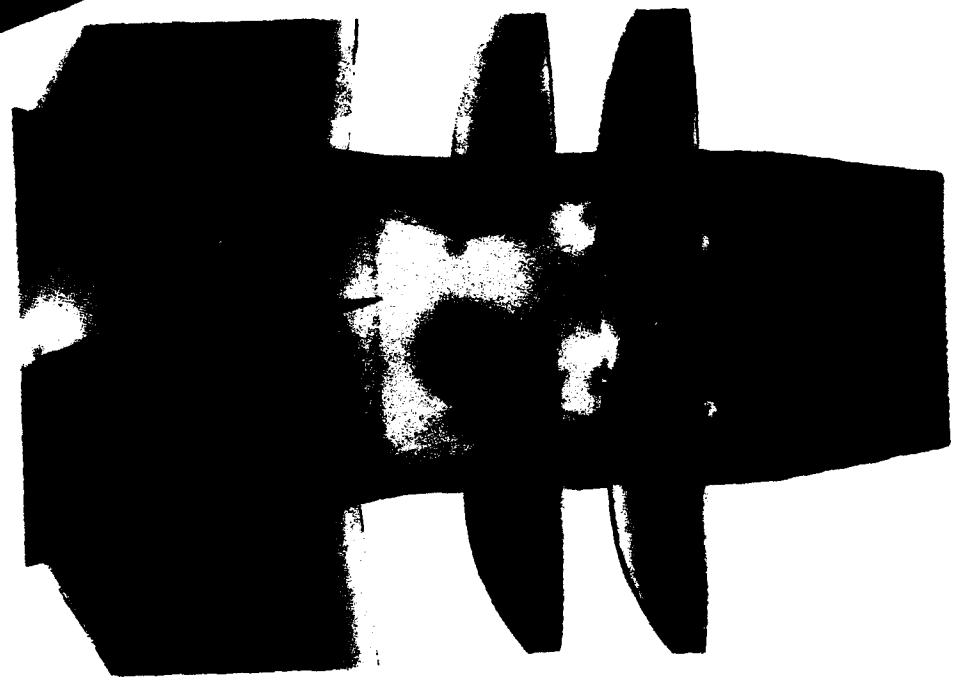


Plate 4.7 Details of aft region for a cruise missile configuration.

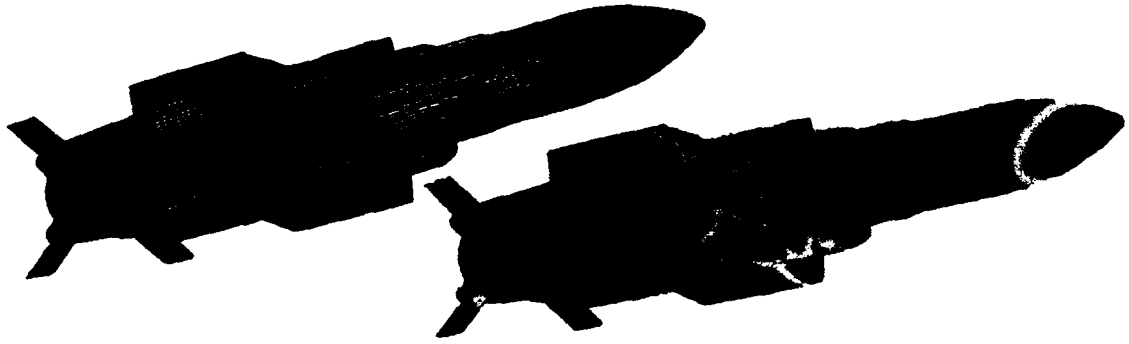


Plate 4.8 Surface grid and pressure coefficient for a missile configuration; $M_\infty = 2.0$, $\alpha = 4.0$ deg.

Plate 4.9 Isobars for pylon-mounted store configuration; split-line 50% between store and wing.

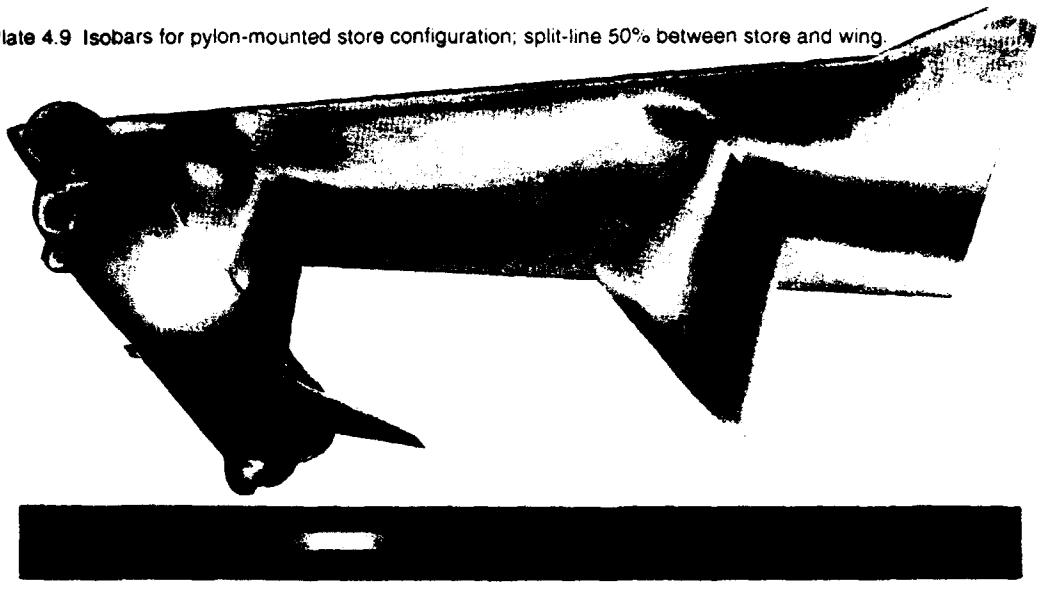
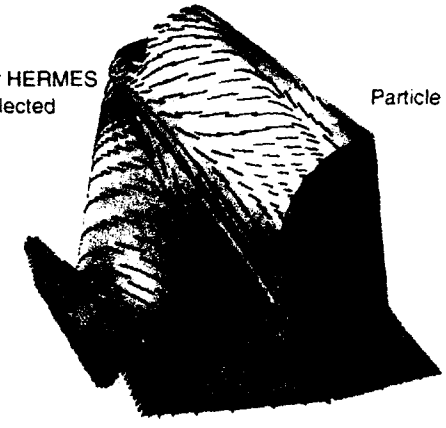


Plate 4.10 Mach contours for HERMES 1.0 configuration with non-deflected flaps.



Particle traces

Plate 4.11 Particle traces for HERMES 1.0 configuration.



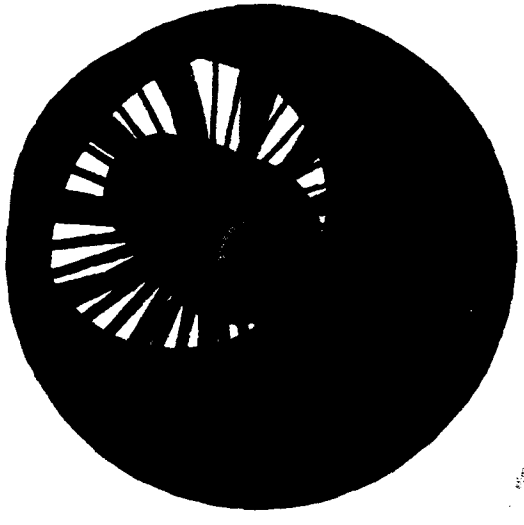


Plate 4.12 Surface pressure contours for Euler computation of fan stage passage.

Concluding Remarks

Our aim has been to give a survey of the state of the art of Computational Aerodynamics Based on the Euler Equations as of 1993. Numerical schemes, algorithms, grid generation, physical and numerical aspects, as well as a wide range of applications have been presented from different points of view.

It has been our intention that the AGARDograph would provide a balanced picture covering fundamental, technical/numerical as well as engineering application aspects. For this reason we have sought for (and found!) a team of authors with different backgrounds ranging from Academia through Research Establishments to Industry. We think the authors did an excellent job in representing the points of view of each of these communities.

We believe that, as reflected in Chapter 1, the physics of inviscid flow and the mathematics of the Euler equations are reasonably well, but not yet fully, understood. Our knowledge on the existence, uniqueness and behaviour of the mathematical solutions of the Euler equations is not complete and the related question of how the Euler equations model flows with separation, circulation and vortices is still subject of discussion. It is suggested that there is (still) room for considering such problems and issues in the more general context of existence and uniqueness of (steady) solutions of the Navier-Stokes equations in the limit of vanishing viscosity. It is also suggested that the choice and the role of the boundary conditions and their numerical implication are crucial in this context.

With respect to numerical schemes and algorithms the picture emerging from chapters 2 and 3 is the following:

- the great majority of codes is based on finite volume (cell-centered or cell-vertex) formulations
- the principle division between codes is between those using (block-)structured grids and those with unstructured grids. The main trade-off factors are efficiency of grid generation and efficiency of flow solver. Although unstructured (adaptive) grid methods seem to gain in importance, in particular for complex geometries and/or for complex flows, block-structured grid methods are, as yet, the most common
- most "production-type" codes for sub/transonic flows use central difference type space discretization. A substantial fraction of codes, in particular those intended for supersonic and hypersonic applications, use some form of upwind discretization technique
- Explicit time stepping schemes of the Runge-Kutta type are characteristic for most codes, in particular those with central difference space discretization. However, implicit features in the form of residual smoothing, have been added in many cases. Other implicit schemes are also represented.
  Convergence acceleration in the form of local time stepping is a feature of all codes. Several if not many production-type codes also use multi-grid (as well as implicit residual smoothing).
  It is the editors impression that in several cases improvements in solver efficiency should be possible. Further research into the possibilities of relatively novel techniques like GMRES and Preconditioning (Chapter 2) is also recommended.

The chapter (4) on Applications illustrates that by now Euler methods are recognized and used in practice as engineering and research tools for the analysis and design of aerospace vehicles in the complete range of speeds from low subsonic to high supersonic and hypersonic. Although different groups of people in the NATO countries have followed different strategies, schemes and algorithms, it seems that all of their methods can provide good results for either specific or more general applications.

The big limitation of Computational Aerodynamics Based on the Euler Equations is, ofcourse, in the absence of modelling of viscous effects at finite Reynolds number. Euler methods are therefore being overtaken rapidly by methods based on the Reynolds-averaged Navier-Stokes equations. This is illustrated by the fact that several of the codes listed in Chapter 3 can be run in Euler as well as in Navier-Stokes mode. (It also leads to the suggestion that the Fluid Dynamics Panel of AGARD should undertake the publication of an AGARDograph on Reynolds-averaged Navier-Stokes methods before the turn of the century!).
It is important to note that proper functioning in Euler mode is a prerequisite for Navier-Stokes codes because of the dominance at high Reynolds numbers of the inviscid, advective terms in the greater part of the flow field. Hence the "Euler technology", subject of this AGARDograph, is equally important for Navier-Stokes codes. It is, in spite of the current shift in emphasis from Euler to Navier-Stokes, also not to be expected that Euler c[illegible] will vanish from the aerodynamicist's tool box in du[illegible]se. The reason is that the computational effort (an[illegible] probably also the manpower) involved with Euler computations is significantly smaller than for Navier-Stokes. This makes Euler methods more attractive for preliminary design studies.
It is in the nature of this AGARD publication that no information is contained about work that has been done in Russia or other non-NATO countries. We probably also missed significant work from people within the NATO community that did not come to our notice. All of those, please accept our apologies.

Finally, we would like to thank again all authors and colleagues from Universities, Research Establishments, and Industries, who contributed and helped to put together this AGARDograph.

Joop Slooff,
Wolfgang Schmidt
Editors,
Fluid Dynamics Panel

REPORT DOCUMENTATION PAGE

1. Recipient's Reference	2. Originator's Reference	3. Further Reference	4. Security Classification of Document
	AGARD-AG-325	ISBN 92-836-1005-9	UNCLASSIFIED

5. Originator Advisory Group for Aerospace Research and Development
North Atlantic Treaty Organization
7 rue Ancelle, 92200 Neuilly-sur-Seine, France

6. Title
Computational Aerodynamics Based on the Euler Equations

7. Presented at

8. Author(s)/Editor(s)	9. Date
J.W. Slooff and W. Schmidt (Editors)	September 1994
10. Author's/Editor's Address	**11. Pages**
See flyleaf	256

12. Distribution Statement There are no restrictions on the distribution of this document. Information about the availability of this and other AGARD unclassified publications is given on the back cover.

13. Keywords/Descriptors

Computational fluid dynamics
Aerodynamics
Euler equations of motion
Fluid flow
Computerized simulation
Numerical analysis
Grids (coordinates)

14. Abstract

A survey of the state of the art of Computational Aerodynamics based on the Euler Equations is presented. For the major Euler Codes that are currently in use in the NATO countries, numerical schemes, algorithms, grid generation, physical and numerical aspects, as well as a wide range of applications are included. Background material required for understanding the physics modelled by the Euler Equations is also presented. Areas of application concentrate on numerical simulation of external flows about aerospace vehicles. Internal flows and turbomachinery applications are not extensively treated but touched upon where considered appropriate.

AGARDograph 325
Advisory Group for Aerospace Research and Development, NATO
COMPUTATIONAL AERODYNAMICS BASED ON THE EULER EQUATIONS
Edited by J.W. Slooff and W. Schmidt
Published September 1994
256 pages

A survey of the state of the art of Computational Aerodynamics based on the Euler Equations is presented. For the major Euler Codes that are currently in use in the NATO countries, numerical schemes, algorithms, grid generation, physical and numerical aspects, as well as a wide range of applications are included. Background material required for understanding the physics modelled by the Euler Equations is also presented. Areas of application concentrate on numerical simulation of external flows about aerospace vehicles. Internal flows and turbomachinery applications are not extensively treated but touched upon where considered appropriate.

ISBN 92-836-1005-9

AGARD-AG-325

Computational fluid dynamics
Aerodynamics
Euler equations of motion
Fluid flow
Computerized simulation
Numerical analysis
Grids (coordinates)

AGARDograph 325
Advisory Group for Aerospace Research and Development, NATO
COMPUTATIONAL AERODYNAMICS BASED ON THE EULER EQUATIONS
Edited by J.W. Slooff and W. Schmidt
Published September 1994
256 pages

A survey of the state of the art of Computational Aerodynamics based on the Euler Equations is presented. For the major Euler Codes that are currently in use in the NATO countries, numerical schemes, algorithms, grid generation, physical and numerical aspects, as well as a wide range of applications are included. Background material required for understanding the physics modelled by the Euler Equations is also presented. Areas of application concentrate on numerical simulation of external flows about aerospace vehicles. Internal flows and turbomachinery applications are not extensively treated but touched upon where considered appropriate.

ISBN 92-836-1005-9

AGARD-AG-325

Computational fluid dynamics
Aerodynamics
Euler equations of motion
Fluid flow
Computerized simulation
Numerical analysis
Grids (coordinates)

AGARDograph 325
Advisory Group for Aerospace Research and Development, NATO
COMPUTATIONAL AERODYNAMICS BASED ON THE EULER EQUATIONS
Edited by J.W. Slooff and W. Schmidt
Published September 1994
256 pages

A survey of the state of the art of Computational Aerodynamics based on the Euler Equations is presented. For the major Euler Codes that are currently in use in the NATO countries, numerical schemes, algorithms, grid generation, physical and numerical aspects, as well as a wide range of applications are included. Background material required for understanding the physics modelled by the Euler Equations is also presented. Areas of application concentrate on numerical simulation of external flows about aerospace vehicles. Internal flows and turbomachinery applications are not extensively treated but touched upon where considered appropriate.

ISBN 92-836-1005-9

AGARD-AG-325

Computational fluid dynamics
Aerodynamics
Euler equations of motion
Fluid flow
Computerized simulation
Numerical analysis
Grids (coordinates)

AGARDograph 325
Advisory Group for Aerospace Research and Development, NATO
COMPUTATIONAL AERODYNAMICS BASED ON THE EULER EQUATIONS
Edited by J.W. Slooff and W. Schmidt
Published September 1994
256 pages

A survey of the state of the art of Computational Aerodynamics based on the Euler Equations is presented. For the major Euler Codes that are currently in use in the NATO countries, numerical schemes, algorithms, grid generation, physical and numerical aspects, as well as a wide range of applications are included. Background material required for understanding the physics modelled by the Euler Equations is also presented. Areas of application concentrate on numerical simulation of external flows about aerospace vehicles. Internal flows and turbomachinery applications are not extensively treated but touched upon where considered appropriate.

ISBN 92-836-1005-9

AGARD-AG-325

Computational fluid dynamics
Aerodynamics
Euler equations of motion
Fluid flow
Computerized simulation
Numerical analysis
Grids (coordinates)

AGARD

NATO OTAN

7 RUE ANCELLE • 92200 NEUILLY-SUR-SEINE

FRANCE

Télécopie (1)47.38.57.99 • Télex 610 176

DIFFUSION DES PUBLICATIONS

AGARD NON CLASSIFIEES

Aucun stock de publications n'a existé à AGARD. A partir de 1993, AGARD détiendra un stock limité des publications associées aux cycles de conférences et cours spéciaux ainsi que les AGARDographies et les rapports des groupes de travail, organisés et publiés à partir de 1993 inclus. Les demandes de renseignements doivent être adressées à AGARD par lettre ou par fax à l'adresse indiquée ci-dessus. *Veuillez ne pas téléphoner.* La diffusion initiale de toutes les publications de l'AGARD est effectuée auprès des pays membres de l'OTAN par l'intermédiaire des centres de distribution nationaux indiqués ci-dessous. Des exemplaires supplémentaires peuvent parfois être obtenus auprès de ces centres (à l'exception des Etats-Unis). Si vous souhaitez recevoir toutes les publications de l'AGARD, ou simplement celles qui concernent certains Panels, vous pouvez demander à être inclu sur la liste d'envoi de l'un de ces centres. Les publications de l'AGARD sont en vente auprès des agences indiquées ci-dessous, sous forme de photocopie ou de microfiche.

CENTRES DE DIFFUSION NATIONAUX

ALLEMAGNE
Fachinformationszentrum,
Karlsruhe
D-76344 Eggenstein-Leopoldshafen 2

BELGIQUE
Coordonnateur AGARD-VSL
Etat-major de la Force aérienne
Quartier Reine Elisabeth
Rue d'Evere, 1140 Bruxelles

CANADA
Directeur du Service des renseignements scientifiques
Ministère de la Défense nationale
Ottawa, Ontario K1A 0K2

DANEMARK
Danish Defence Research Establishment
Ryvangs Allé 1
P.O. Box 2715
DK-2100 Copenhagen Ø

ESPAGNE
INTA (AGARD Publications)
Pintor Rosales 34
28008 Madrid

ETATS-UNIS
NASA Headquarters
Code JOB-1
Washington, D.C. 20546

FRANCE
O.N.E.R.A. (Direction)
29, Avenue de la Division Leclerc
92322 Châtillon Cedex

GRECE
Hellenic Air Force
Air War College
Scientific and Technical Library
Dekelia Air Force Base
Dekelia, Athens TGA 1010

ISLANDE
Director of Aviation
c/o Flugrad
Reykjavik

ITALIE
Aeronautica Militare
Ufficio del Delegato Nazionale all'AGARD
Aeroporto Pratica di Mare
00040 Pomezia (Roma)

LUXEMBOURG
Voir Belgique

NORVEGE
Norwegian Defence Research Establishment
Attn: Biblioteket
P.O. Box 25
N-2007 Kjeller

PAYS-BAS
Netherlands Delegation to AGARD
National Aerospace Laboratory NLR
P.O. Box 90502
1006 BM Amsterdam

PORTUGAL
Força Aérea Portuguesa
Centro de Documentação e Informação
Alfragide
2700 Amadora

ROYAUME-UNI
Defence Research Information Centre
Kentigern House
65 Brown Street
Glasgow G2 8EX

TURQUIE
Millî Savunma Başkanliği (MSB)
ARGE Dairesi Başkanliği (MSB)
06650 Bakanliklar-Ankara

Le centre de distribution national des Etats-Unis ne détient PAS de stocks des publications de l'AGARD. D'éventuelles demandes de photocopies doivent être formulées directement auprès du NASA Center for Aerospace Information (CASI) à l'adresse ci-dessous. Toute notification de changement d'adresse doit être fait également auprès de CASI.

AGENCES DE VENTE

NASA Center for
AeroSpace Information (CASI)
800 Elkridge Landing Road
Linthicum Heights, MD 21090-2934
Etats-Unis

ESA/Information Retrieval Service
European Space Agency
10, rue Mario Nikis
75015 Paris
France

The British Library
Document Supply Division
Boston Spa, Wetherby
West Yorkshire LS23 7BQ
Royaume-Uni

Les demandes de microfiches ou de photocopies de documents AGARD (y compris les demandes faites auprès du CASI) doivent comporter la dénomination AGARD, ainsi que le numéro de série d'AGARD (par exemple AGARD-AG-315). Des informations analogues, telles que le titre et la date de publication sont souhaitables. Veuiller noter qu'il y a lieu de spécifier AGARD-R-nnn et AGARD-AR-nnn lors de la commande des rapports AGARD et des rapports consultatifs AGARD respectivement. Des références bibliographiques complètes ainsi que des résumés des publications AGARD figurent dans les journaux suivants:

Scientific and Technical Aerospace Reports (STAR)
publié par la NASA Scientific and Technical
Information Division
NASA Headquarters (JTT)
Washington D.C. 20546
Etats-Unis

Government Reports Announcements and Index (GRA&I)
publié par le National Technical Information Service
Springfield
Virginia 22161
Etats-Unis
(accessible également en mode interactif dans la base de données bibliographiques en ligne du NTIS, et sur CD-ROM)

Imprimé par le Groupe Communication Canada
45, boul. Sacré-Cœur, Hull (Québec), Canada K1A 0S7